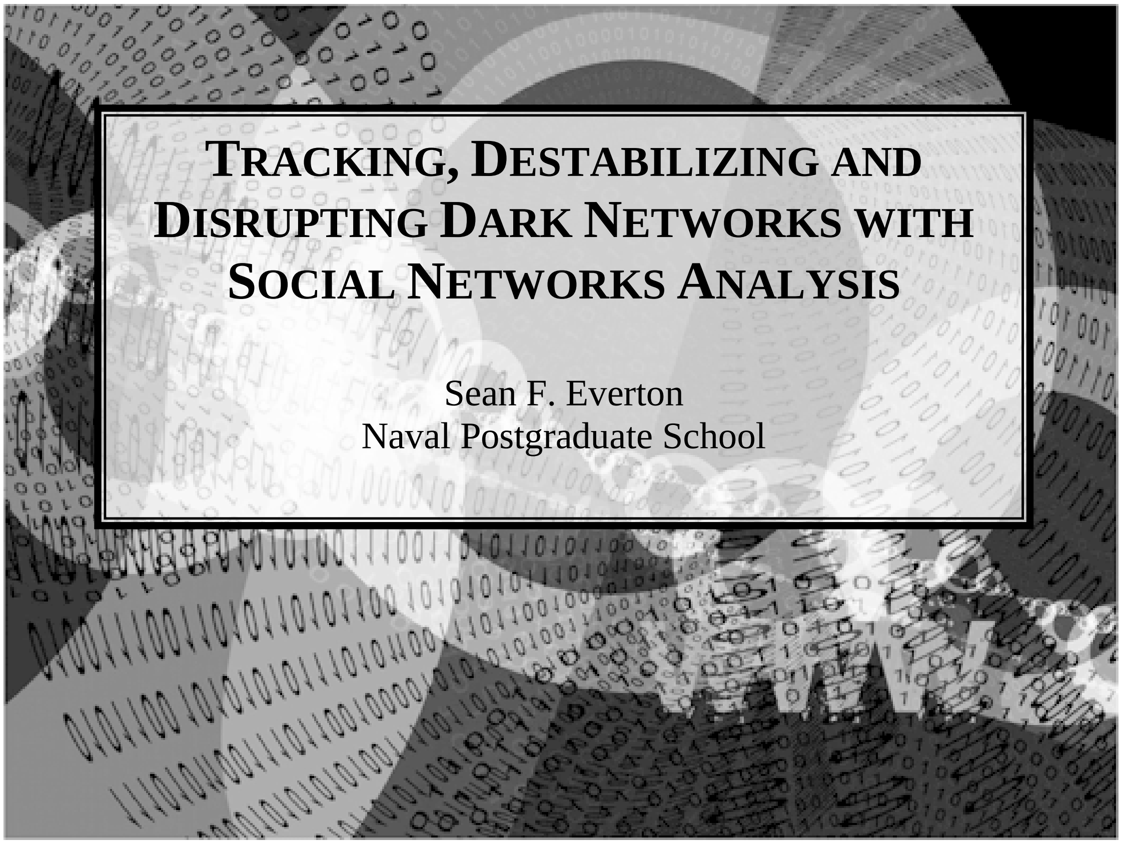


TRACKING, DESTABILIZING AND DISRUPTING DARK NETWORKS WITH SOCIAL NETWORKS ANALYSIS

Sean F. Everton
Naval Postgraduate School

Version 1.05

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

This guide introduces how to use social network analysis to develop strategies for tracking and disrupting dark networks (i.e., criminal, terrorist networks). While much of its focus is on social network analysis, it is more than a methodological handbook. It includes a strategic emphasis as well. It is one thing to learn how to estimate basic social network analysis metrics. It is quite another to learn how to place such metrics within a larger strategic framework. That is why this introduction begins by briefly outlining how we understand social network analysis's strategic role for destabilizing and disrupting networks. (It is a topic that we turn to in more detail in the final chapter.) After this outline, it turns to the unique challenges that dark networks pose for analysts, including those employing social network analysis. It concludes with an overview of what is to come in the following chapters.

1.1 Social Network Analysis and Dark Networks: A Strategic Emphasis

Social network analysis can be used in both exploratory and confirmatory (i.e., hypothesis testing) ways. Exploratory social network analysis "involves visualization and manipulation of concrete networks, whereas hypothesis testing boils down to numbers representing abstract parameters and probabilities" (de Nooy, Mrvar and Batagelj 2005:xxv). Presently, exploratory social network analysis dominates the discipline and is the primary way we use it in this guide. Specifically, we attempt to demonstrate how to use exploratory social network analysis for developing strategies (i.e., hypotheses) for destabilizing and disrupting dark networks. The general process for doing this is as follows:

- What is your working hypothesis (i.e., what is your question)?
- Identify and record relationships of interest
- Aggregate networks based on working hypothesis
- Analyze network using metrics of interest
- Interpret/discuss your findings (What do they mean?)
- Develop strategies and making policy recommendations for tracking, monitoring, disrupting the network

While we focus on each of these steps in more depth later in the guide, it is appropriate to discuss each of them briefly before proceeding.

Developing a Working Hypothesis

Before attempting to use social network analysis, researchers need to first develop a working (i.e., tentative) hypothesis as to how best to disrupt the dark network they are analyzing. Some, for example, may decide that the best approach is to focus on the network's financial ties, believing that by shutting down or disrupting the flow of funds, it will be difficult for the network to finance operations. Others may conclude that targeting a network's operational ties offers the best opportunity for disruption, assuming that without its key operatives, a network will not have the personnel necessary to carry out operations. Still others may target a network's friendship, kinship, school and/or religious ties since insurgencies tend to recruit new members through close personal ties (Lofland and Stark 1965; Sageman 2004b; Snow, Zurcher and Ekland-Olson 1980; Stark and Bainbridge 1980). Some may choose not to focus on a network's individual ties at all but instead on its institutional ties, arguing that a network can most effectively be disrupted by focusing on the groups/organizations that helped give rise to and currently sustain the insurgency (Smith 1996). In short, there are multiple ways of tracking and disrupting dark networks, of which these are only a few examples. The broader point here is that before beginning to collect data on a dark network, researchers need to first develop working hypotheses as to how they plan to disrupt the network.

Identifying, Collecting and Recording Social Network Data

Working hypotheses guide the types of relationships researchers will eventually collect and record. At the individual level these can be friendship ties, school ties, operational ties, religious ties and so on. At the group level, they can be ties between two institutions because they share a common member.[1] That said collecting social network data can be a tedious and complicated process. Not only do we have to determine which ties are important, we have to determine where the network begins and ends. That is, we have to determine its boundaries. We may decide to analyze a particular network, but identifying which actors are members of the network and which ones are not is generally easier said than done.

As we discuss in more detail later (see Chapter 4), social network analysts generally record network data in matrix form. Take, for instance, the following

[1] For example, if someone teaches at a particular school and attends a local community of faith, one could argue that a tie exists between the school and the community of faith.

subset of marital ties between Renaissance Florentine families collected and recorded by John Padgett and Christopher Ansell (1993) and used by Breiger and Pattison (1986). A tie was determined to exist if a member of one family married a member of another family. The data are recorded in a square matrix with a row and column for each family. If a marriage occurred between two families, then a "1" appears in the families' common cells (e.g., Acciaiuol & Medici).[2] If not, a "0" appears. As we will see later, there are instances where ties are not reciprocal and are recorded accordingly.

	ACCIAIUOL	ALBIZZI	BARBADOR	BISCHERI	CASTELLAN	GINORI	GUADAGNI	LAMBERTES	MEDICI	PAZZI	PERUZZI	PUCCI	RIDOLFI	SALVIATI	STROZZI	TORNABUON
ACCIAIUOL	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
ALBIZZI	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0
BARBADORI	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
BISCHERI	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0
CASTELLAN	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0
GINORI	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GUADAGNI	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	1
LAMBERTES	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
MEDICI	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	1
PAZZI	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
PERUZZI	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0
PUCCI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
RIDOLFI	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1
SALVIATI	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
STROZZI	0	0	0	1	1	0	0	0	0	0	1	0	1	0	0	0
TORNABUON	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0

Figure 1.1: Marriage Ties Among Renaissance Florentine Families

As one can imagine, when working with very large networks, recording network data in matrix form can prove challenging and why social network analysts often turn to other methods when working with large networks. For our purposes here, however, we focus on recording data in matrix form.

Aggregating Networks

Actors are typically involved in more than one type of relation (Hanneman and Riddle 2005). For example, most individuals are embedded in a number of different types of ties, such as friendship, kinship and economic. Business organizations are no different. They engage in financial and informational exchanges and sometimes form alliances with one another (Saxenian 1994). The same is true with countries. They are explicitly linked through numerous cultural, economic, military and political ties, and implicitly linked through ties created by transnational corporations, nongovernmental organizations and international agencies (Meyer et al. 1997). Such multiplexity is important because ties often pull actors in different directions (Simmel [1908, 1922] 1955). For example, our work ties may push us in the direction of making one choice, while our friendship and kinship ties may pull us in another. This suggests that if we want a more

[2] Because marriage ties are reciprocal, a "1" not only appears in the Acciaiuol-Medicci cell, but also in the Medicci-Acciaiuol cell.

accurate picture of a network's dynamics, the more types of ties we record (and aggregate together) the better. "Indeed it is a basic assumption of those subscribing to the network approach that behavior cannot be explained in terms of any one single activity" (Breiger 1975, cited in Azarian, 2005:39).

This, of course, raises questions of strategy and the coding of social network data. If we want to disrupt a dark network's financial operations, we may want to only aggregate its financial ties (e.g., ties of financial flows between actors) and actor's affiliations with institutions that provide financial support to the dark network (e.g., businesses, criminal activities, and/or charities and foundations acting as a financial front to the network). But, what if a network's financial operations rely heavily on ties of trust between actors? Then we may want to aggregate friendship and kinship ties along with a network's financial ties. Or, we may want to first analyze only the network's financial ties and then add the friendship and kinship ties for additional analysis. Clearly, there are no clear-cut answers as to how and which networks to aggregate. Such decisions have to be made on a case-by-case basis.

Analysis: Metrics and Visualization

Social network metrics play an important role in analyzing a social network's dynamics. Density and other related measures can help researchers gain an overall understanding the overall "shape" of the network (i.e., its topography); centrality measures can help identify key and peripheral actors within a network; clustering algorithms can help locate various subgroups within the larger network (and also provide additional information on the network as a whole); and brokerage measures can help identify actors and ties between actors that serve as channels for the exchange and flow of information and other resources (e.g., financial, affection). Social network analysts generally use a variety of metrics (rather than just one or two) in their attempts to gain an overall understanding of a network.

Metrics are not the only tool available to social network analysts, however. Network visualization is another helpful tool that it can help us see patterns that may not be readily apparent by simply looking at metrics (Castilla et al. 2000). Indeed, sometimes when clustering algorithms are unsuccessful in identifying cohesive subgroups within a network, network visualization algorithms are. That is why both metrics and visualization are complimentary parts of the social network analysis toolbox. Most social network analysis programs either come with network mapping algorithms (e.g., Pajek) or integrate with network visualization programs that do (e.g., UCINET and NetDraw).

Interpretation

Once we finish analyzing a network, we need to interpret what we have found. We will want to ask questions such as:

- Does the network as a whole exhibit any characteristics (e.g., density, centralization, clustering) that might suggest whether it is more or less effective than other comparable networks?
- Are there key actors or ties between actors whose removal from the network will render it ineffective?
- Or, are there peripheral actors that could possibly be enticed to leave the network, thus making it more vulnerable to disruption, infiltration or isolation?
- Are there distinct subgroups within the larger network that could possibly be turned against one another such that the network exerts more energy on internal fights than external operations?
- Are there organizations within the network (e.g., schools, businesses, faith communities) that are attractive targets for infiltration to improve our intelligence gathering capabilities?

Needless to say, the answers we give to these questions and others inform the strategies we develop and the policies we recommend.

Developing Strategies and Policy Recommendations

It is important to emphasize that the strategies we develop and the policies we recommend do not necessarily have to involve direct action (e.g., targeting key actors within the network for removal or capture). They could instead involve indirect action (IO and Psyops) campaigns that manipulate actors and/or relationships within the network, rendering it less effective, or amnesty campaigns that attempt to reintegrate some network members (e.g., individuals or faith communities) back into the larger community, thus weakening the network from the outside. Or, they could involve targeting key institutions within the network for infiltration, thus improving our intelligence gathering (and providing better network data). The point here is that social network analysis provides various ways of looking at a dark network, which in turn provides us with potentially different strategies for disrupting them.

Summary

It is not an overstatement to state that the bottom line of this approach to using social network analysis to track and disrupt dark networks is to develop strategies and make related policy recommendations. That is why this guide is much more than a methodological handbook. It is also helpful to keep in mind that social network analysis is only one tool available to researchers. It is not a magic bullet that will single-handily win the global war on terror. It is, however a valuable tool that when used in conjunction with other tools (e.g., geospatial analysis, temporal analysis, cultural analysis) can prove to be quite valuable.

1.2 Dark Networks as Problems

Many argue that because of their adaptability, network structures (as opposed to hierarchical ones) are better suited to solving nonroutine, complex and/or rapidly-changing "problems" or challenges (Granovetter 1985; Raab and Milward 2003). For instance, AnnaLee Saxenian (1994; 1996) argues that Silicon Valley emerged as the center of the high technology universe because it developed a highly-flexible industrial network—characterized by a horizontally integrated industrial system, flat corporate structures, friendly local institutions, a supportive culture and a networked institutional infrastructure—that was more responsive to the volatile high technology industry than were other regional areas.

Of course, for some the problem to be "solved," however, is not always how to best develop a regional high-tech economy but rather how to safely engage in covert and illegal activities such as arms trafficking, drug dealing, terrorism or money laundering (Raab and Milward 2003). It appears that network structures are well suited for tackling these types of problems as well (Klerks 2001; Krebs 2001; Raab and Milward 2003).

This creates certain problems for those who seek to disrupt them. One is that because of their network structure, covert and illegal (i.e., dark) networks are quick to adapt to changing environmental pressures. For instance, prior to the September 11th attacks Al Qaeda was somewhat vertically integrated, at least at the command and control level; since the U.S. invasion of Afghanistan, however, it appears to have become far more decentralized (Raab and Milward 2003:425). This suggests that network data at "time one" may be out of date at "time two," so we continuously need to update our data. Another problem is that dark networks do not necessarily operate independently from one another but instead are often connected through actors who function as brokers between these networks (Raab and Milward 2003:425):

> A truism of the network approach is that, at some level, everything is connected to everything else. This is no less true of dark networks. There is increasing evidence of a close connection between Al Qaeda and the failed states of Liberia, Sierra Leone, and Burkina Faso in West Africa. The connection appears based on Al Qaeda's need to exchange cash for diamonds. This is fueled by the pressure from the United States and Western Europe to clamp down on Al Qaeda's use of legitimate banks for international monetary transactions. Diamonds provide a ready currency for Al Qaeda, and the failed states of the region have perhaps provided a safe haven for Al Qaeda's operatives in the wake of 11 September in exchange for arms and money for the warlords of the region.

Consequently, as we discussed earlier correctly specifying a network's boundaries is of the upmost importance. Misspecification could lead to the development of inappropriate strategies and recommendations. Finally, because dark networks are, by definition, "dark" (i.e., they actively seek to keep their activities secret), collecting the relational data needed for social network analysis is something of a challenge. "Although it is often difficult for scholars to get sufficient relational data in the case of overt networks, it might, in many instances, be impossible in the case of covert networks" (Raab and Milward 2003:435). Put simply, analysts are constantly faced with the possibility that our data are incomplete. This speaks to the importance of considering the adoption of strategies that improve our intelligence gathering capabilities. To be sure, these three problems—incompleteness of data, fuzzy boundaries, and dynamic and evolving networks (Krebs 2001; Sparrow 1991)—are not unique to dark networks. They arise with light networks as well. It is just that with dark networks, they are more acute.

Does that mean we should abandon the social network approach for disrupting dark networks? Not at all. Given that a large number of covert and illegal groups appear to have adopted a network form of organization, it seems incumbent that we use a theoretical and methodological tool specifically designed for analyzing social network dynamics. Moreover, the network approach's insistence on the "systematic collection of information about the relations among the social units" (Raab and Milward 2003:435), can only improve our understanding of the dark network. Thus, while we should not shy away from using social network analysis for analyzing dark networks, we need to always keep in mind the limitations of the approach.

1.3 Preview of Coming Attractions

This guide's organization reflects its strategic emphasis. It is structured in such a way that it not only introduces researchers to basic social network theories and techniques, but also embeds these theories and techniques in the larger strategic framework that is crucial for tracking, destabilizing and/or disrupting dark networks. This introductory chapter and the next seek to provide readers with a foundation on which to use social network methods for analyzing dark networks. While this chapter introduces readers to the strategic context of social network analysis as well as provides an overview of what is to come, Chapter 2 introduces the basic terms, concepts and assumptions of social network theories and methods. Some may conclude that a theoretical discussion, however brief, is unnecessary, but what this chapter attempts to demonstrate is that we cannot conduct good social network analysis apart from knowledge of the theories and assumptions lying behind the various methods.

The next two chapters provide an introduction to some of the basic skills needed for social network analysis. Chapter 3 seeks to help users become comfortable with the three social network analysis software packages that we use in this guide: UCINET,[3] the "granddaddy" of social network analysis software programs; NetDraw,[4] a program for visualizing social network data developed by the same people who create UCINET; and Pajek,[5] a social network analysis package that integrates metrics and visualization. Chapter 4 focuses on the basics of collecting, recording, manipulating and visualizing social network data.

Chapters 5 through 8 examine various social network methodologies: Chapter 5 looks at a variety of metrics for getting a sense of how the network is structured as a whole; Chapter 6 explores how various measures of centrality can be used to identify a network's key and peripheral players; Chapter 7 examines a variety of methods for detecting subgroups within the larger network; and Chapter 8 explores methods for identifying actors and ties that broker the flow of information and other resources within and through the network.

The final chapter returns to a central topic of this first chapter and places these various social network methods into the larger strategic context of how to

---

[3] UCINET 6.0 (Borgatti, Everett and Freeman 2002) can be purchased from Analytic Technologies at www.analytictech.com.
[4] NetDraw (Borgatti 2002-2005) comes as part of the UCINET 6.0 package but can also be downloaded separately at the Analytic Technologies website: www.analytictech.com.
[5] Pajek (Batagelj and Mrvar 2007) is a network analysis and graph drawing program designed to handle extremely large data sets that can be downloaded for free for noncommercial use from the Pajek web site: http://vlado.fmf.uni-lj/pub/networks/pajek.

track and disrupt dark networks, highlighting not only direct approaches but indirect approaches as well. It considers various perspectives in order to illustrate how we can think of social network analysis in a more macro context. This final chapter does not purport to be exhaustive, only illustrative. Analysts are free to integrate social network analysis with other perspectives as they see fit.

It is important to note that this guide does not provide a comprehensive introduction to all the various theories and techniques associated with social network analysis. Instead, it seeks to bridge the gap between theory and practice by demonstrating how to apply various theories and methods to specific examples.[6] It typically begins with examples from small networks in order to illustrate the theory and method, and then moves to a more complex example, namely, the terrorist network of Noordin Mohammed Top (International Crisis Group 2006). Indeed Top's network serves as a "running" example throughout the guide, from the initial collection of social network data to estimating various social network analysis metrics to strategies for disrupting dark networks in general (see Appendix 1 for a more complete discussion of the data source).

This guide has also adopted the approach used by Everton (2004) de Nooy, Mrvar and Batagelj (2005) and Hanneman and Riddle (2005) in that it not only discusses various social network techniques and metrics, it illustrates how to estimate them using UCINET, NetDraw and Pajek.[7] Because these programs are regularly updated, there is a good chance that the various dialog boxes, command menus and report windows illustrated in this manual will not always match what readers encounter when working with these programs. Nevertheless, most changes should be minor and should not cause the readers too much difficulty.

[6] Readers who are interested in general introductions to social network analysis should consult Scott (2000), Degenne and Forse' (1999) and/or Knoke and Yank (2007). Wasserman and Faust (1994) offer a more comprehensive (and mathematical) introduction. Robert Hanneman and Mark Riddle (2005) have written a helpful introduction to social network analysis methods using UCINET, while Walter de Nooy, Andrej Mrvar and Vladimir Batagelj (2005) have done the same for those interested in Pajek.

[7] Following de Nooy, Mrvar and Batagelj (2005) these commands are placed in the margin next to the text discussing the technique/metric in order to make them easier to find.

CHAPTER 2

INTRODUCTION TO SOCIAL NETWORK ANALYSIS

What is social network analysis? From where did it develop? Scott (2000) traces the roots of social network analysis to the work of gestalt and sociometric theorists, such as Fritz Heider, Kurt Lewin, and Jacob Moreno (Heider 1977; Lewin 1951; Moreno 1953), who emphasized how organized patterns shape how we see and interpret the world; and the work of social anthropologists, such as Siegried Nadel (1957) and Alfred Radcliffe-Brown (1940), whose ideas about the relationship between social patterns and social structure influenced later researchers, such as Elton Mayo (1933; Mayo 1945; see also Roethlisberger and Dickson 1939), W. Lloyd Warner (Warner and Lunt 1941), John Barnes (1954), Elizabeth Bott (1957) and J. Clyde Mitchell (1969). While the work of these individuals failed to create an identifiable social network paradigm, leading social network analysis to endure a "dark ages" of sorts (Freeman 2004:63-120), they laid the groundwork for its renaissance at Harvard in the 1960s and 1970s (Freeman 2004; Scott 2000), which was led by Harrison White and his students (see, e.g., Granovetter 1973, 1974; White 1992, 2008; White, Boorman and Breiger 1976).[8] The discipline has been blossoming ever since (Freeman 2004). Social network analysts have created their own organization (the *International Network for Social Network Analysis*), they have launched their own journals (*Connections*, *Social Networks* and the *Journal of Social Structure*), they gather annually in either North America or Europe (*Sunbelt meetings*), and they have produced a number of standard texts on social network analysis (Knoke and Yang 2007; Scott 2000; Wasserman and Faust 1994). With the recent entry of physicists and other scientists into the field (Barabasi 2002; Barabasi and Albert 1999; Barabasi, Albert and Jeong 1999; Barabasi and Bonabeau 2003; Buchanan 2001, 2002; Watts 1999a, 1999b, 2003), social network analysis has become something of a cottage industry.

How does social network analysis differ from more traditional approaches (i.e., variable-based) to analyzing data? The basic difference is that while variable-based approaches focus on account actors' attributes (e.g., gender, race,

[8] Other traditions that have informed social network analysis (but often not acknowledged by social network analysts) are exchange theory (Cook and Whitmeyer 1992; Emerson 1972a, 1972b, 1976) and research into the recruitment of individuals to religious and social movements (Gould 1991, 1993; Lofland 1977; Lofland and Stark 1965; McAdam 1986, 1988; Snow and Phillips 1980; Snow, Zurcher and Ekland-Olson 1980).

education) and generally ignore the broader social interaction patterns in which they are embedded (e.g., at home, work and place of worship), social network analysis focuses on how these interaction patterns affect behavior, noting that while many attributes remain the same across social contexts, most interaction patterns do not, which suggests that interaction patterns are just (or perhaps more) important for predicting and understanding behavior than are attributes (Knoke and Yang 2007).

> A woman who holds a menial job requiring little initiative in an office may be a dynamic leader of a neighborhood association and an assertive PTA participant. Such behavioral differences are difficult to reconcile with unchanging gender, age and status attributes, but comprehensible on recognizing that people's structural relations can vary markedly across social contexts (Knoke and Yang 2007:5).

Social network analysis, then, is a collection of theories and techniques that provide empirical content to social context. It has been used successfully to explain varieties of behavior because it forces researchers "to think in terms of constraints and options that are inherent in the way social relations are organized" (Raab and Milward 2003). For example, Padgett and Ansell (1993) found that whether or not certain elite families in 15^{th}-century Florence supported the Medici's or one of its rival political factions depended more on the pattern of economic, patronage and marital ties than on the various families' class and status attributes (Knoke and Yang 2007:5).

The rest of this chapter is devoted to introducing readers to the basic terms, concepts and assumptions of social network analysis. Some readers may be tempted to skip this chapter, but unless you are well-versed in social network analysis, that would be a mistake. Gaining an understanding of social network analysis's basic terms and concepts is necessary so that you do not become lost in over the course of later discussions. Of course, not all terms and concepts are discussed here. Others will be introduced as the guide progresses, and Appendix 2 provides a glossary of the all the various terms and concepts that you encounter in this guide. A theoretical discussion of social network analysis is also necessary because social network analysis cannot appropriately be done apart from a grasp of the assumptions lying behind it. To be sure, with today's computers and software anyone can issue a command and estimate a metric, but without a genuine understanding of the assumptions lying behind these methods, you may very well choose the wrong one.

2.1 Basic Terms and Concepts

Actor

In social network analysis the term actor refers to discrete individuals, subgroups, organizations, collectivities, communities, nation-states, and so on that are involved in social relations. As this suggests social network analysis does not always focus at the individual level, a fact that has largely been ignored by analysts using social network analysis in their attempts to disrupt dark networks. Within social network analysis, sometimes actors are referred to as *nodes* and *vertices*.

Tie

Actors are linked together by ties. Ties vary in both their type and strength. Examples of types of ties include (Wasserman and Faust 1994:18)

- Ties of sentiment (friendship, liking, respect)
- Resource ties (business transactions, financial flows)
- Ties of association or affiliation (members of the same church, club, etc.)
- Behavioral ties (communication ties)
- Ties based on geographic movement (migration, physical mobility)
- Ties based on status movement (social mobility)
- Ties based on physical connection (road, river or bridge connecting two points)
- Formal ties (organizational hierarchy)
- Biological ties (kinship)

Ties vary on a continuum from strong to weak (Granovetter 1973, 1974). At the individual level, we can think of strong ties as those where actors have repeated and relatively intense interactions with one another, whereas we can think of weak ties are actors who see one another occasionally or rarely. Nevertheless, it is not always self-evident where the cut-off between a strong and weak tie exists (Krackhardt 1992). Moreover, what distinguishes a weak tie from the numerous, random and usually unrepeated encounters actors experience on a daily basis is not always clear (Azarian 2005:37). Determining a threshold or cut-off value for identifying what constitutes a tie and what does not is (or at least should be) a difficult task. Thus, it is helpful to think of a social tie as “a theoretical construction, abstracted by the analyst from the bulk of largely erratic streams of

affections, encounters and interactions between a pair of actors, be they human beings, informal groups, formal organizations, or others" (Azarian 2005:37).

Social Network (and Social Network Analysis)

A social network is simply a finite set or sets of actors that share ties with one another (Wasserman and Faust 1994:21), while social network analysis involves the detection and interpretation of the patterns of social ties among actors (de Nooy, Mrvar and Batagelj 2005:5). Figure 2.1 depicts a hypothetical social network. The ellipses (i.e., circles) represent actors while the lines (solid and broken) represent relations or ties. As this hypothetical network suggests, seldom are actors located randomly in networks; instead, they typically cluster within relatively distinct subgroups. Moreover, some actors are generally embedded relatively deeply within a subgroup, while others sit more on the periphery, serving as bridges between subgroups.

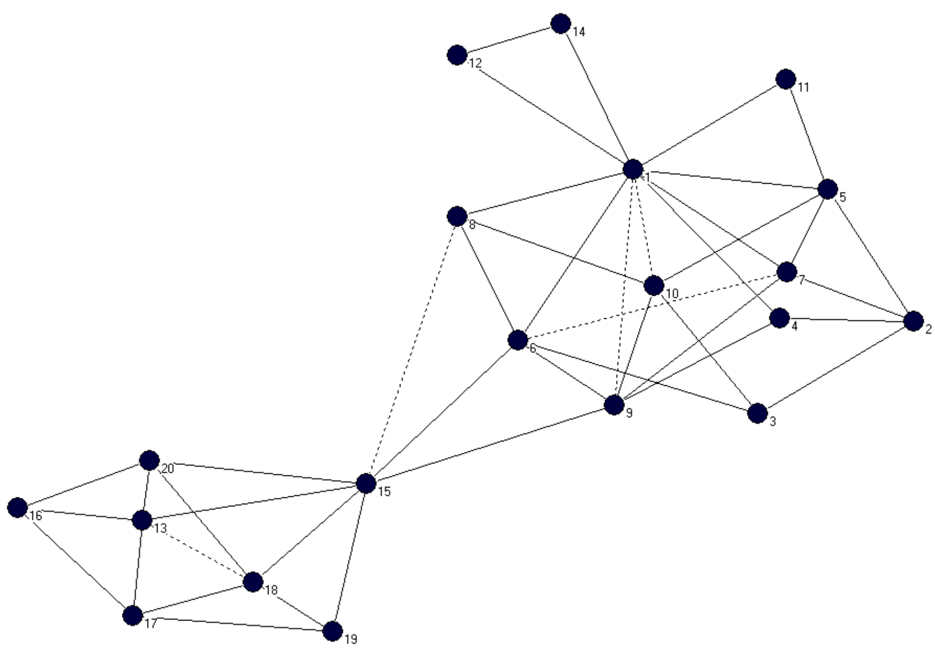


Figure 2.1: Hypothetical Social Network

Density

Network density is a characteristic of a network as a whole. Formally it is defined as the total number of ties within a network divided by the total possible number of ties. Consequently, network density measures can theoretically range from 0.0 to 1.0, such that in a network with a density of 0.0 no ties exist between actors whereas in a network with a density of 1.0 all possible ties exist between

actors. For example, the hypothetical network displayed in Figure 2.1 above contains 44 lines out of a possible of 190 lines; thus its density is .2316. Of course, a social network with no ties is, by definition, not a social network, and seldom (except in very small networks) do we discover networks where all possible ties are actually present. Nevertheless, an awareness of a measure's limits can be helpful in grasping the assumptions lying behind it. We will consider network density and other related measures in Chapter 5.

Researchers have found that network density is positively related to the likelihood that actors within the network will follow accepted norms and behavior, which is why a primary basis for moral order is highly-connected social networks. Why? One reason is that in dense networks it is easier for people to monitor the behavior of others and prevent them from engaging in deviant behavior (Granovetter 2005). Another is that most people are more likely to conform to social norms when they run the risk of losing their relationships to others (Finke and Stark 2005), and in dense networks we are more like to have ties (relationships) that we do not want to lose. In sparse networks, however, we often lack the social ties that would otherwise prevent us from misbehaving. Take frontier areas like the Wild, Wild, West, for instance. People are constantly passing through, which makes it hard for social ties to form, so social networks tend to be sparse. Sparse networks also make it difficult for institutions (like churches) to form, which is why frontiers tend to short on piety and long on deviance (Finke and Stark 2005).

Related to this is the phenomenon that some call the law of group polarization (Sunstein 2003:111-144). It predicts that when like-minded people deliberate as an organized group, the general opinion shifts toward extreme versions of their common belief (Bauerlein 2004). In a product-liability trial, for example, if nine jurors believe the manufacturer is somewhat guilty and three believe it is entirely guilty, the latter will draw the former toward a larger award than the nine would allow on their own. Or, if people who object in varying degrees to the war in Iraq convene to debate methods of protest, all will emerge from the discussion more resolved against the war. Sageman's (2004b) study of the global salafi jihad uncovered similar group dynamics. He found that people who joined the jihad were often homesick young men who drifted to familiar settings, like mosques, to find companionship and alleviate their loneliness. There, small clusters of friends formed. They often moved into apartments together where they underwent a long period of intense social interaction in their apartments and developed strong mutual intimacy (i.e., formation of dense networks). As they became closer, they progressively adopted the beliefs and faith of their most extreme members. This distanced them further from their

childhood friends and family, leading to increased isolation and loyalty to the group, which in turn intensified their faith, and they were then ready to join the jihad.

Path (and Path Distance)

Notions of *path* and *path distance* are probably easier to illustrate than define, so here we provide both a definition and an illustration. A *path* is defined as a *walk* (i.e., a sequence of actors and ties) in which no actor in-between the first and last actor of the walk occurs more than once, whereas the *path distance* between two actors is the number of steps between the two actors. Looking at Figure 2.1 above you can trace a path from actor 9 to actor 19 through the actor 15, and the path from actor 6 to actor 11 through actor 1. In both cases the distance between the actors is two (i.e., two steps). It is quite common for there to be numerous paths between actors to exist, some longer and shorter than others. The shortest path between two actors is known as a *geodesic*.

Centrality

Notions that certain actors are more central than others go back at least as far as Jacob Moreno's (1953) conception of sociometric stars and isolates (de Nooy, Mrvar and Batagelj 2005). Alex Bavelas (1950) was the first to formally investigate the properties of centrality as he looked at how a network influences the flow of communication in experimental groups (2000). Not surprisingly, most social networks contain people or organizations that are more central than others and because of their position, they often enjoy better access to information and better opportunities to spread information. Social network analysts have identified several measures of centrality, each based on different assumptions of what it means to be more central. The most commonly used are degree, closeness, betweenness and eigenvector.

- Degree centrality is the count of the number of an actor's ties
- Closeness centrality measures (based on path distance) how close, on average, each actor is to all other actors in a network
- Betweenness centrality measures the extent to which each actor lies on the shortest path between all other actors in a network
- Eigenvector centrality assumes that ties to highly central actors are more important than ties to peripheral actors, so weights an actor's summed ties to other actors by their centrality scores

We will consider these measures in more depth in Chapter 6.

Cohesive Subgroups

A major focus of social network analysis is to identify dense clusters of actors “among whom there are relatively strong, direct, intense, and/or positive ties” (Wasserman and Faust 1994:249). Social network analysts often refer to these clusters of actors as cohesive subgroups and generally assume that “social interaction is the basis for solidarity, shared norms, identity, and collective behavior, so people who interact intensively are likely to consider themselves a social group” (de Nooy, Mrvar and Batagelj 2005: 61). Social network analysts use several approaches for identifying cohesive subgroups. One way is to cluster actors based on attributes (e.g., race, gender, etc.). Another is to focus on the pattern of ties (i.e., relations) among actors. In an ideal world there would be one method that we could use to identify cohesive subgroups. However, we do not live in an ideal world, so it should not come as a surprise that social network analysts have developed a variety of methods for identifying clusters of actors (Scott 2000). Chapter 7 explores some of the various approaches for using patterns of ties for identifying cohesive subgroups within social networks.

Attributes

While social network analysis focuses primarily on the pattern of ties between actors, it does not completely ignore attribute data, which are characteristics of the individual actors. If you are talking about individuals, then this is information such as gender, race, ethnicity, years of education, income level, age, etc. If the actors in a social network are corporations, then attribute variables can measure total sales, net income, age of the corporation, number of employees, and so on. And if you are talking about countries, then attribute variables might measure GDP per capita, population size and so on. Finally, as we will see later in Chapter 6, centrality measures (once calculated) become attributes of actors as well.

Sometimes the boundary between attributes and affiliations can be somewhat fuzzy. As a general rule, something is an affiliation if two actor’s participation in that affiliation indicates a relationship, but it is also possible for an affiliation to function as an attribute as well. For example, some members of Noordin Top’s terrorist network have Afghanistan military experience. If they were in Afghanistan at the same time, this affiliation may indicate a tie between the two. At the same, military experience in Afghanistan could function as an indicator of status (and thus an attribute) within Noordin’s terrorist network.

2.2 Assumptions

While some have noted that social network analysis is more of a method than a theory (see e.g., Granovetter 1979), most social network analysis methods are built on a common set of assumptions (Azarian 2005; Knoke and Yang 2007; Wasserman and Faust 1994):

- Actors and their related actions are interdependent, rather than independent, with other actors
- Ties between actors are seen as channels for the transfer or flow of various types of resources (e.g., funds, information, trust, enmity, etc.)
- Social structures are seen in terms of enduring patterns of ties between actors
- An actor's position in the social structure (i.e., its structural location) impacts its beliefs, norms and observed behavior
- Social networks are dynamic entities that change as actors, subgroups, and ties between actors enter or leave the network.

Each of these assumptions is discussed in turn. We illustrate them with examples from various studies not only to make them more intelligible but also to draw out possible implications for applying social network analysis to dark networks.

Interdependence of Actors

Social network analysis assumes that actors do not make decisions as independent autonomous units but instead are strongly influenced by the behavior and choices of other actors. At the individual level this can be illustrated by Solomon Asch (1955) conformity experiments and Stanley Milgram's (1974) obedience to authority experiments.

Solomon Asch: Social Conformity

In his social conformity experiment, Solomon Asch sorted college students into groups of 8 to 10 and told them that they were participating in a study about visual perception. The experiment entailed 18 trials in which two cards, similar to those below (Figure 2.2, next page), were projected on a screen. Asch instructed the students that they were to choose the bar on the right card that was the same length as the bar on the left card. Moreover, they were state their answers out loud so that all of the other participants could hear their answer.

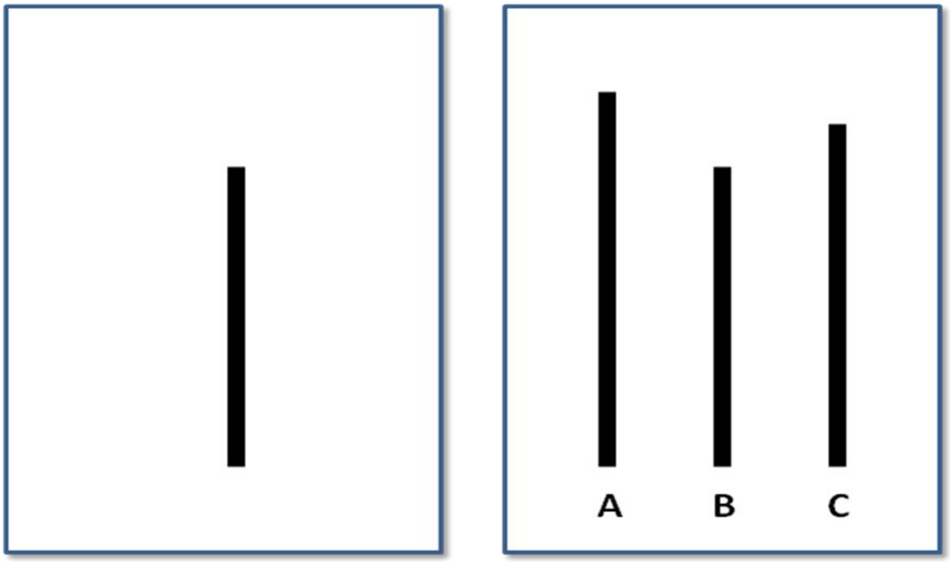


Figure 2.2: Solomon Asch's Conformity Experiment

Of course, the experiment was rigged. Only one of the students was a real subject – the rest were Asch's confederates, who gave incorrect answers on 12 of the 18 trials. Asch made sure that the real subjects were the next-to-the-last person to announce their answer so that they would hear most of the confederates' incorrect responses before they gave their own. He was curious to see whether the subjects would feel any pressure to give the same answers as the confederate majority even when the latter clearly answered incorrectly. Asch found that 37 of the 50 subjects conformed to the majority at least once, while 14 conformed on more than 6 of the 12 trials. The average conformity rate was 1/3 (4 out of 12 trials). Asch varied the number of confederates from 1 to 15; he found that subjects conformed to a group of 3 or 4 as readily as they did to larger groups. He did find that subjects were less likely to conform if they had an ally, however. In some of his experiments, Asch told one of his confederates to give the correct answer. When he did this, people conformed ¼ as often as they did in the original experiment.

Stanley Milgram: Obedience to Authority

Subjects who participated in Stanley Milgram's "Obedience to Authority" experiments were told that the experiment was designed to test the effect of punishment on learning. Upon arriving they were sorted by the drawing of slips into "teachers" and "learners." Unbeknownst to the subjects, the drawing of slips was set up in such a way that the true subjects always ended up as teachers and Milgram's confederates always ended up as learners. Milgram (or one of his experimenters) told the teachers (i.e., the subjects) that their job was to teach a

series of word pairs to the learner; their specific task was to administer a shock to the learner each time he made a mistake recalling a word. The teachers watched as the learners were strapped into an electric chair with an electrode taped to their wrist that the teachers were told was attached to a shock generator. The teachers were then taken to an adjacent room and seated in front of the shock generator. The generator had 30 switches, indicating that the number of volts that could be administered, ranging from 15 to 450, accompanied by labels ranging from "Slight Shock" to "Very Strong Shock" "Danger: Severe Shock" to "XXX" (evidently indicating a lethal level of shock). The experimenter then told the teachers that they were to increase the shock level by 15 volts each time the learners gave a wrong answer.

The learners, of course, did not really receive any shocks, but the teachers (i.e., the subjects) did not know this. Initially, the learners only voiced verbal protests about the painfulness of the shocks, but once the shock level reached 300 volts, they began pounding the wall and, from that point on, gave no answers. After awhile, the pounding even stopped. Throughout the experiment, the experimenter would restate the teachers' duties. If the teachers looked to the experimenter for guidance, the experimenter would say, "Please continue." If they protested that the learners were not answering, the experimenter would state that the learners' failure to answer should be treated as a wrong answer. If the teachers expressed reluctance to continue or suggested that the learners' condition should be checked, the experimenter would insist that "the experiment requires that you continue." If the teachers became really insistent, the experimenter would say, "You have no choice; you must go on."

In the end, every single teacher (i.e., subject) went beyond the 300 volt level and more than half—65%—obeyed to the 450-volt bitter end. These results shocked a great number of people, and not just those participating in the experiment. Up until this point many people believed that the Holocaust was a product of German culture or psyche, but these experiments suggested that ordinary individuals will, in certain social contexts, do horrible things to their fellow human beings. This is especially true when people are asked to obey people they perceive to be authorities or experts in their field (Sunstein 2003:35). In a variation on the experiment where the experiment was conducted in a seedy hotel room and administered by a student wearing cutoff jeans with boxers showing, only 20% of the people obeyed to the end.

Implications

These studies and others (Zimbardo, Maslasch and Haney 2000) suggest that far from acting independently of those around them, people in fact do just the

opposite. In the face of peer pressure, Asch's student subjects chose to go along with the crowd even when the correct answer was obvious. How much more likely are people to go along with the crowd when they are presented with much more ambiguous information? Milgram's subjects demonstrate how perceived expertise or authority can lead people to make choices that one would hope they would otherwise not make. How are we to effectively combat the global salafi jihad (Sageman 2004a, 2004b) when members of terrorist networks look to "respected" authorities such as Osama bin Laden and Ayman al-Zawahiri for inspiration? Put simply, these studies suggest that when analyzing the behavior of actors, if we do not take into account the social context in which they are embedded, we could arrive at a serious misunderstanding of their actions.

Following the crowd is not limited to individuals. As John Meyer, Woody Powell and Paul DiMaggio (and their numerous colleagues) have repeatedly pointed out (see e.g., DiMaggio and Powell 1983; Frank, Hironaka and Schofer 2000; Meyer et al. 1997; Meyer and Rowan 1977; Powell and DiMaggio 1991), groups, corporations and nation-states are no more likely to act autonomously than are individuals. For example, organizations that interact with one another tend to, over time, become more like one another. This tendency is not driven primarily by concerns over the bottom line, but by the concern that these organizations maintain their legitimacy in the eyes of other similar organizations (DiMaggio and Powell 1983). "When an organizational practice or structure becomes commonly understood as a defining feature of a 'legitimate' organization of a certain type, organizational elites feel pressure to institute that practice or structure. If there is a cultural norm that says, 'In order for an organization to be a good organization, it must have characteristic X,' organizations feel pressure to institute characteristic X" (Chaves 1997:32-33). Consequently, when analyzing dark networks, not only should we pay attention to the networks of individual actors, we should pay attention to networks of institutional actors as well.

Ties as Channels

Another assumption of social network analysis is that ties (i.e., relations) between actors can function as channels for the transfer or flow of various types of resources (e.g., information, funds, affection, and trust). For example, Mark Granovetter's (1973; 1974) study of how people found their present jobs demonstrated how "weak" ties can function as conduits of job information. Specifically, he discovered that people were far more likely to have used personal contacts in finding their present job than by other means. While approximately

19% used formal means[9] to find their current job and another 19% directly applied for their job,[10] approximately 56% used personal contacts. Moreover, of those who found their jobs through personal contacts, most were "weak ties" (i.e., acquaintances, not close friends). He also discovered that only 16.7% said that they saw their contact regularly at the time they heard about the job, while 55.6% said they saw their contact occasionally, and 27.8% said rarely (Granovetter 1973:1371). Moreover, workers who were not job hunting when they found their present jobs were more likely to have heard about them through weak ties. All this led Granovetter to argue that when it comes to finding jobs, our weak ties – that is, our acquaintances – can be quite useful.

Why? Because our acquaintances (i.e., our weak ties) are less likely to be socially involved with one another than are our close friends (i.e., our "strong ties"). Thus, the set of people making up our network of acquaintances tends to be relatively sparse, while the set of people making up our network of close friends tends to be relatively dense.

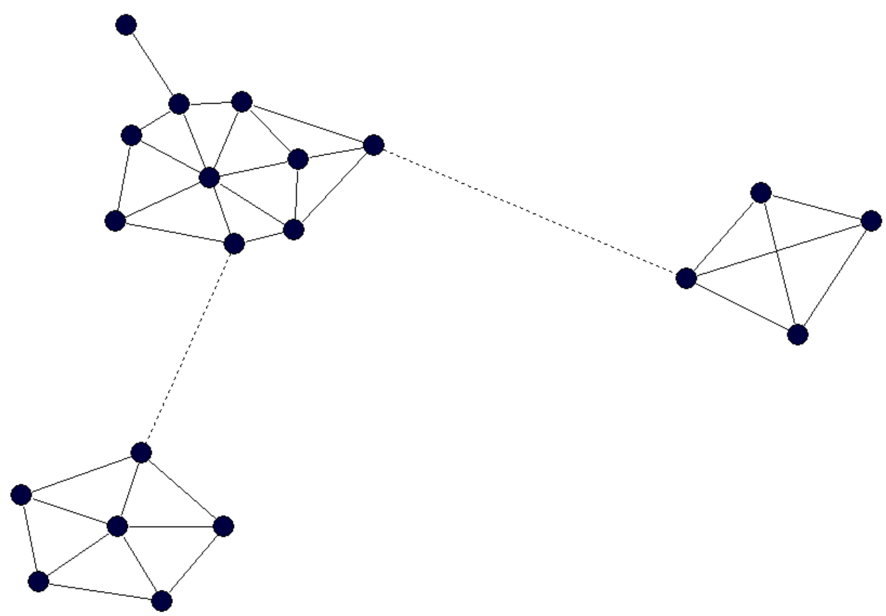


Figure 2.3: Strong and Weak Ties

Imagine the pattern of social ties suggested by this argument (Figure 2.3) and take any individual in the network. He or she will most likely have a collection of close friends, most of whom know one another. This same

[9] Where the job seekers used the services of impersonal intermediaries such as advertisements, public and private employment agencies, interviews and placements sponsored by universities or professional associations

[10] Where the job seekers went or wrote directly to a firm, did not use a formal or personal intermediary, and had not heard about a specific opening from a personal contact

individual will also probably have a collection of acquaintances, few of whom know one another. But each of these acquaintances, in turn, is likely to have close friends in their own right, so they are also likely to be embedded in a tightly knit network of their own but one different from our original individual. According to Granovetter weak ties are important in terms of the overall structure of a network because they form the crucial bridges that tie together densely knit clusters of people. In fact, if it were not for these weak ties, these clusters would not be connected at all.

Granovetter (1973:1363) also argues that while not all weak ties are bridges, all bridges are weak ties. This is because of what he calls the forbidden triad (Figure 2.4).

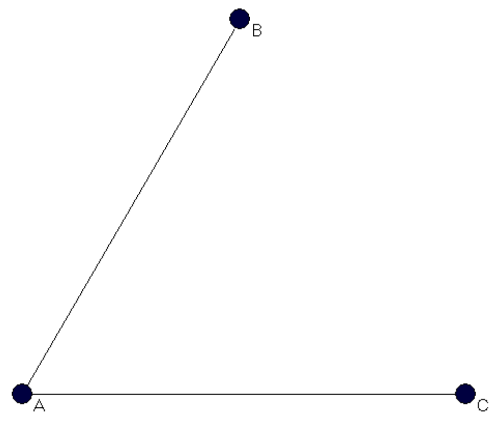


Figure 2.4: Granovetter's Forbidden Triad

Imagine that the ties between A & B and A & C are strong and that initially B and C have no relationship with one another. In the short run the strong ties that run from C to B through A will function as a bridge between C & B, but according to Granovetter, eventually a tie will form between C & B. Why? Because since the ties between A & B and A & C are strong, odds are that in the long run B & C will get to know one another and a tie will form between them, and the strong ties running between C & B through A will no longer function as a bridge between C & B. While Granovetter admits that this notion of the forbidden triad is something of an exaggeration, research suggests that this holds true most of the time.

Implications

A primary implication of Granovetter's argument is that whatever resource is to be diffused – whether it is job information, influence, resources, etc. – it will reach a larger number of people and traverse greater social distance when it passes through weak ties rather than strong ones. Another implication is that actors with

few weak ties are more likely to be deprived of information from distant parts of the social system than are actors with many weak ties. As he puts it, "instead, they will be confined to the provincial news and views of their close friends" (Granovetter 1973). Their lack of weak ties will not only insulate them from the latest ideas and fashions, but it may also put them at a disadvantage in the labor market where knowing about appropriate job openings at just the right time is paramount.

Does this have any implications for dark networks? Yes. In his analysis of the March 11, 2004, Madrid bombings, Rodriguez (2005) found that weak ties were a key feature of the terrorist network in that they enabled its cells to main operative ties with the larger network from which they were able to draw material supplies and ideological support. Rodriguez also believes that weak ties provide benefits to dark networks in other ways. He argues, for instance, that weak ties provide dark networks with (1) relative stability when members are arrested or missions fail, (2) more flexibility that allows them to rapidly adapt to a changing environment and (3) higher levels of security because weak ties are harder to detect than strong ones.

This is not to say that strong ties are of no value. Indeed, "there is a mountain of research showing that people with strong ties are happier and even healthier because in such networks members provide one another with strong emotional and material support in times of grief or trouble and someone with whom to share life's joys and triumphs" (Stark 2007:37). Thus, feelings of trust and solidarity are more likely to be shared across strong ties than across weak ones.

This suggests that in order for dark networks to operate effectively they require an optimal mix of weak and strong ties. It also suggests that if we want to disrupt a dark network, altering its mix of weak and strong ties may be a viable strategy. We will return to this topic in Chapter 5.

Social Structure in Terms of Social Networks

Social scientists frequently refer to the concept of social structure. By this term they usually have in mind the enduring patterns of behavior and relationship within social systems (e.g., roles) or to social institutions and norms that have become embedded in social systems in such a way that they shape behavior (Wikipedia 2007). Take, for example, the following passage from an introductory sociology textbook:

> If you stand on the sidelines and watch the world pass you by, what you see is people moving about and somehow avoiding each other, people talking, people

> going in and out of buildings, people sitting on benches, people driving cars, people congregating, and in general you see just a maze of activity as individuals move about in physical space. There is fluidity to social life when examined this way, but there is also order, at least most of the time. People are not randomly moving about, talking, driving, entering and exiting buildings, or sitting around; they have purposes and goals as they move in space and talk to each other. But there is more than just purpose; there is structure to what you see as you look at the ebb and flow of human activity. Part of this structure inheres in the organization of symbols into culture, as it directs and guides individuals to act in certain ways. But for culture to be really effective in regulating conduct, it must be attached to something that orders social life. This extra "something" is social structure. Social structures constrain who is present, where they stand, what they can do, and how they are related to each other. This structure is as real as the buildings that people occupy (Turner 2006: 88).

However, while social structure may be as real as the buildings that people occupy, it is difficult to measure empirically. It is easy to talk about social structure in the abstract, but it is difficult to quantify. What social network analysis does is provide analysts with a way of systematically and empirically studying the causes and consequences of social structure (Degenne and Forse' 1999). It is a collection of theories and methods designed to explore the patterns of ties between people, groups of people, organizations and countries.

Structural Location: Beliefs, Norms and Behavior

Social network analysis assumes that actors' structural location has important perceptual, attitudinal and behavioral implications. Our attitudes, our beliefs and so on are largely determined by our location in the social structure. We can illustrate how this is so by examining what social scientists have learned about the process of conversion (recruitment) to religious and social movements. Be advised, however, that this is not the only instance of social action that is affected by structural location.

Conversion and Recruitment: Moonies, Mormons and Movements

For years deprivation theory was the reigning theory of conversion. It argued that people join particular groups because they suffer from some sort of deprivation that the group's ideology (or theology) addresses. Put differently, religious groups meet particular needs – whether they are economic, social or political. Researchers would look at a group's ideology to see what kinds of

deprivation it addressed, and then conclude that people who joined that group must have been suffering from that sort of deprivation. For instance, since Christian Science promises to restore health, social scientists might conclude that Christian Science draws disproportionately from people suffering from chronic health problems (Glock 1964, cited in Stark 1996:15). Of course, you could also argue that only people with good health would join and stay with a faith that believes that illness is only a state of the mind (Stark 1996:15). While it makes perfect sense that some people will find a particular ideology or theology more appealing than others, this does not explain why only some of these ideologically or theologically suited people actually join. That is where social network theory comes in**.** The basic insight of social network theory is that people will choose one religious group over another (or none at all) based on their patterns of friendship. People tend to join groups where they know people. They tend not to join groups where they don't know anyone.

Take, for instance, John Lofland and Rodney Stark (1965; see also, Stark 1996:13-21) study of people converting to the Moonies. A woman named Miss Kim, who had come to California from Korea (via Oregon) where she had been a university professor, started the local Moonies group that they observed. When she first arrived, she spoke at a number of public events, but these did not yield her any converts. Instead, her first three were close friends of hers who she first got to know after she became a lodger with one of them. Next, some of the husbands joined, who were then followed by friends from work. The next converts were old friends, relatives or people who first formed close friendships with one or more members in the group. As Stark notes, when he and Lofland first began watching, "the group had never succeeded in attracting a stranger" (Stark 1996:16). They observed a number of people who were sympathetic with the group's doctrines, but in the end did not join because they had numerous social ties with people who disapproved of the Moonies. This led them to conclude that the people who ultimately joined the Moonies tended to be those whose social ties to group members exceeded their social ties to nonmembers (Stark 1996:16).

In another study, Rodney Stark and Bill Bainbridge (1980) looked at the role that social ties play in recruiting people to the Mormon Church. Mormons tend to keep very good records of their missionary efforts and which methods work better than others, and Stark and Bainbridge were provided with data for all missionaries in the state of Washington during 1976-1977 (Stark and Bainbridge 1980:1386). Mormons recruit through a variety of means: they go door to door, they follow up on referrals and they meet potential recruits in the home of a relative or friend of the potential recruits. Interestingly, when missionaries go door to door, their success rate is only .1%. Referrals provide a somewhat higher rate of success (7%

for covert referrals and 8% for overt referrals). Their highest rates of success, however, occurred when Mormons invite non-Mormon friends and relatives into their homes to meet Mormon missionaries. In those instances, they enjoy a success rate close to 50%. This suggests that the best strategy for conversion is not cold-calling but forming friendships with non-Mormons. Stark and Bainbridge note that an article in the Mormon Church's official magazine provided detailed instructions on how to recruit new members, and a recurring theme in these instructions is the importance of building close personal ties with non-Mormons. It explicitly instructed its readers that they should downplay or avoid discussing religion while forming these ties. Only later are they to bring up that they are Mormons (Stark 2005:79-80). "Another way of looking at these findings is that missionaries do not serve as the primary instrument of recruitment to the Mormon faith. Instead, recruitment is accomplished primarily by the rank and file of the church as they construct intimate interpersonal ties with non-Mormons and thus link them into a group network" (Stark and Bainbridge 1980:1387-1388).

Shortly after the Stark and Bainbridge study appeared, a study by David Snow and a few of his colleagues (Snow, Zurcher and Ekland-Olson 1980) highlighted essentially the same dynamic: successful social movements, religious or otherwise, recruit primarily through social networks of friends and families. All of the groups they studied, except the Hare Krishna, recruited over 50% of their members through either kinship or friendship networks with several recruiting over 90% of their members through such networks. The lone exception was the Hare Krishna's. Why? Because Hare Krishna religious groups demand exclusive participation from their members and require them to sever all extra-movement ties. Thus, they have no social networks outside of the group from which they can draw new recruits, which forces them to recruit from public places. This is also why they are so small. Social movements must maintain open social networks in order to grow.

Implications

Conceived of in terms of social network analysis, what these studies tell us is that people do not join groups randomly. Instead recruitment and conversion occurs through social ties. People who are located socially proximate to a group are far more likely to join that network than are those who are socially distant. Put differently, the structural location of actors is a large factor in determining which groups they will join and which ones they will not. Thus, it is not surprising that Sageman (2004a; 2004b) found that recruitment to the global jihad occurs primarily through social ties. 65% of members had pre-existing friendship ties

with someone in the group, while another 15% had kinship ties. After eliminating the overlap of friendship and kinship ties, Sageman found that 75% of the terrorists he studied joined through friendship and kinship ties. Another 8% of the recruits had ties to teachers who had links to the jihad, indicating that 83% of the individuals who joined the global jihad, joined through some sort of social ties. This has led Sageman to argue that while factors such as anger at U.S. policies may increase the pool of potential recruits to the Global Salafi Jihad, only those who have a link to it actually join.

Dynamic Social Networks

Finally, social network analysis assumes that networks are dynamic entities that change as actors, subgroups, and ties between actors enter and leave. For example, a key actor's departure from a network can disrupt the flow of information within the network, whereas the departure of a network's peripheral players could lead the network to turn in on itself. On the other hand, an actor's entrance into a network could temporarily destabilize the network as new lines of communication and trust are negotiated among network members. All this suggests that not only do researchers need to closely monitor changes in dark networks but also that careful analyses will often suggest ways to disrupt or destabilize them. This is illustrated by Mark Sageman's analysis of the potential vulnerabilities of the global salafi jihad, which builds upon recent research into the small world phenomenon.

Stanley Milgram, Small Worlds and Scale-Free Networks

Some years ago Stanley Milgram conducted an experiment to estimate the average number of steps separating two randomly selected individuals (Travers and Milgram 1969). When he first began to publicly speak about the experiment, he would often ask how many steps people thought it was. They typically estimated the number to be in the hundreds. The result, however, was closer to six, and it led to the phrase "six degrees of separation." The idea that on average only six intermediaries separate any two individuals in the world is fairly well known in today's world. Indeed the term, "six degrees of separation" has become part of our cultural lexicon.

Different (and competing) theories exist as to why the world is so "small." One theory argues that the social world is small because individuals tend to cluster into tightly-knit groups and the overall average path length (i.e., distance) between them is relatively short (Watts 1999a, 1999b, 2003). This theory, if correct, has implications for adopting strategies for disrupting and/or destabilizing dark

networks. We only mention it here but will return to it in Chapter 5. Another theory argues that the social world is small because it exhibits the characteristics of a "scale free network" (Barabasi 2002; Barabasi and Albert 1999; Barabasi and Bonabeau 2003). Scale-free networks are networks where a handful of actors have many connections (i.e., hubs), but most actors have very few. The internet is an example of a scale-free network; most internet routers have relatively few connections, but a few have hundreds of thousands. Because most actors in a scale-free network have very few ties, scale-free networks are relatively immune to random failures. That is why the internet seldom, if ever, crashes. The random failure of internet routers simply does not cause too much disruption because they have so few connections. Scale-free networks are vulnerable to targeted attacks, however; studies indicate that the entire internet could crash if 10-15% of its routers were disabled.

Implications

In his research on terrorist network Mark Sageman (2004a; 2004b) found that the global salafi jihad exhibits the characteristics of a scale-free network. That is it contains a handful of actors within the network are hubs but most are not. This led him to initially argue that the United States should focus its efforts on taking out hubs rather than randomly stopping terrorists at our borders. "[The latter] may stop terrorists from coming here, but will leave the network undisturbed. However… if the hubs are destroyed, the system breaks down into isolated nodes. The jihad will be incapable of mounting sophisticated large scale operations like the 9/11 attacks and be reduced to small attacks by singletons" (Sageman 2004a). Needless to say, the simultaneous removal of 10-15% of dark network's hubs is easier said than done, and subsequent research suggests that hubs are quickly replaced by another, highly central and/or structurally equivalent actor (Pedahzur and Perliger 2006; Tsvetovat and Carley 2005). Recently, Sageman (2008) appears to have backed off somewhat from the viability of targeting hubs. Nevertheless, his initial argument provides a good illustration of how one could possibly use overall network structure to develop a strategy for disrupting a dark network.

2.3 Summary and Conclusion

This chapter has provided a brief overview of the basic terms and concepts of social network analysis along with five key assumptions that underlie much, if not all, of what social network analysts do. These assumptions are often more implicit in their work than explicit, but nevertheless they are there (or at least one

hopes they are). Needless to say, they provide the foundation on what we cover in later chapters. For instance, the various centrality metrics that we will examine provide analysts with a sense of actors' structural locations. Similarly, measures of density and network clustering attempt to capture the overall structure of a social network. We will not always draw explicit ties between these assumptions and specific metrics, but hopefully they should become more and more obvious the longer you immerse yourself in social network analysis.

CHAPTER 3

GETTING STARTED WITH UCINET, NETDRAW AND PAJEK

The development of the personal computer has played a large role in the development of social network analysis. Indeed, it is unlikely that it could not have developed without them (Freeman 2004:139-141; Wolfe 1978) for the simple reason that social network analysis relies heavily on complex mathematical and graphical algorithms (Freeman 2004:3, 135-136). Over the years a number of social network software packages have been developed, all of which have their own strengths and weaknesses (see Huisman and van Duijn 2005 for a comprehensive review of available packages). In this guide we focus on UCINET, NetDraw and Pajek not because they are necessarily the best, but because they are readily available, widely used and relatively inexpensive. Users who gain a familiarity with the basic logic and features of these programs should be able to easily learn other social network software packages if necessary.

3.1 UCINET

UCINET was initially developed by Linton Freeman at the University of California, Irvine (hence, the "UCI" in "UCINET") and later refined by others, in particular, Steve Borgatti and Martin Everett (Scott 2000:178-179). It is the best known and most widely used social network software, primarily because it has been around for awhile and it contains a large number of social network analysis metrics and data management tools (Huisman and van Duijn 2005:275-280). For example, not only does it contain most of the routines needed for estimating measures of network topography (e.g., density, clustering, transitivity, etc.), calculating actor centrality (e.g., degree, betweenness, closeness, eigenvector, etc.) identifying subgroups (e.g., cliques, components, factions, etc.) and estimating various measures of structural equivalence (e.g., structural, automorphic, regular, etc.), it also includes tools for selecting subsets of files, merging and stacking data sets, transposing and/or recoding data, and the importing and exporting of data in a variety of formats (Huisman and van Duijn 2005).

UCINET's comprehensiveness is what makes it so valuable. Not only can you find and estimate most of the metrics you will need for social network analysis, if you can record social network data in UCINET format, you can get it into just about any other format you may need for analyzing with other software programs (e.g., Pajek, R, Excel, SPSS). Moreover, UCINET is constantly being

updated by its developers, adding new routines and fixing any bugs or glitches than come to light.

UCINET stores social network data in matrix format. Users can enter data using either UCINET's internal spreadsheet function or a commercial spreadsheet program such as Microsoft Excel, which can then be imported (or pasted) into UCINET. UCINET also reads edge and node lists, which are quite useful when working with large datasets or when data are stored in database programs such as Microsoft Access. In storing data UCINET uses a dual file system: one containing the actual data (extension .##d) and one containing information about the data (extension .##h). Because you need both files in order to analyze social network data in UCINET, this dual file system leads to occasional problems. For example, when estimating routines that create new data files, UCINET will sometimes store one the two newly-created files in separate folders, making it impossible to open or analyze the data until the two files are reunited. Because UCINET's creators regularly provide updates to the program in order to fix bugs and glitches, problems such as this are generally corrected relatively quickly. Nevertheless, they can be quite irritating when they happen. A related problem occurs when analysts share social network data with one another and send only one of the two files to the other. All this is to highlight how important it is when working with UCINET to be aware of its dual file system.

Getting Started with UCINET

When you open UCINET you encounter an interface similar to the one displayed in Figure 3.1 (see next page). Across the top are a series of menus (i.e., File, Data, Transform, Tools, Network, Visualize, Options and Help) – in a moment we will explore some of the commands found under these menus that are useful to know before trying to do too much with UCINET. Other commands (e.g., commands for estimating various centrality measures) we will postpone until later. Just below the menus are a handful of speed buttons. Ranging from left to right, the first speed button closes UCINET, the second opens UCINET's internal spreadsheet program, the third opens Microsoft's "Notepad" program (useful for editing some data files), the fourth allows users to display a social network file in matrix format, while the fifth opens NetDraw, a social network drawing program that reads and displays UCINET files (see discussion of NetDraw in section 3.2 below). Let us now turn to a discussion of some of the commands found under UCINET's various menus.

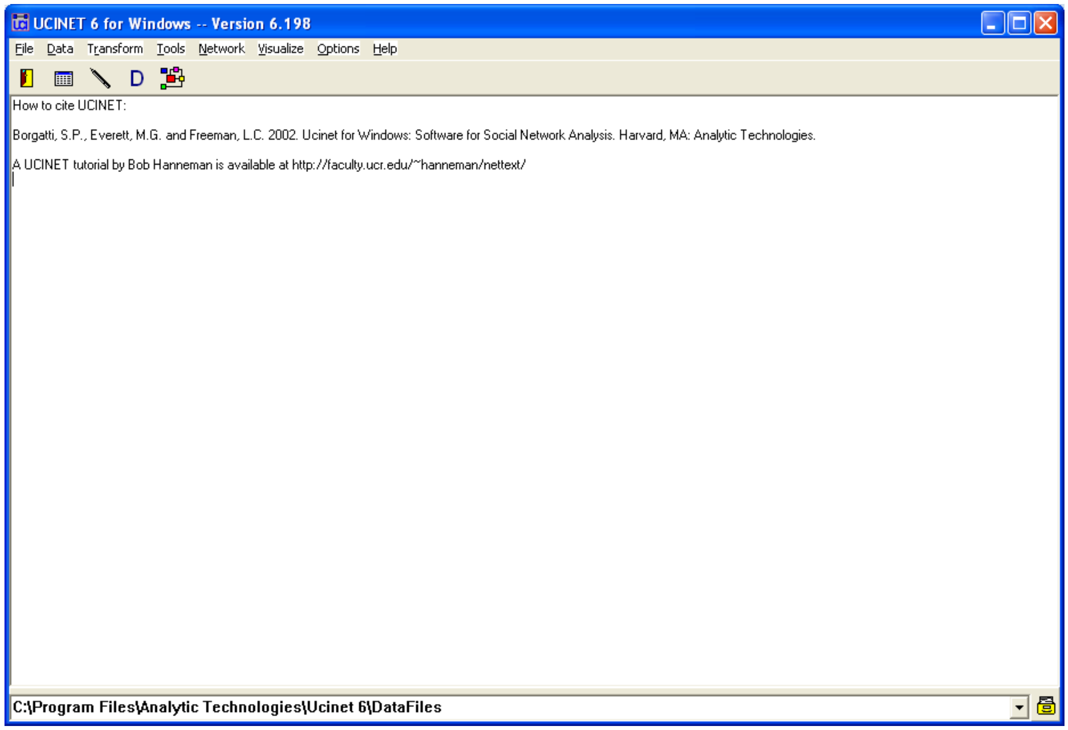


Figure 3.1: UCINET Interface

File >Change Default Folder

The *File* menu contains several useful functions, two of which we will discuss here. One is the *File>Change Default Folder* command, which tells UCINET where to look for social network data.[11] When you use UCINET the first time, the default folder should be "C:\Program Files\Analytic Technologies\Ucinet 6\DataFiles" (note that UCINET lists the default folder at the bottom of its interface – see Figure 3.1), which is where all of the standard social network software datasets that come with UCINET are stored. For now, leave this as your default folder. Later, when working with your own data, you will probably want to change it to the folder where your data is stored. Issuing this command brings up a dialog box that allows you to navigate through your computer's folders in order to pick the default folder of your choice. Another command worth noting here is the *File>Text Editor* command; it opens Microsoft's "Notepad" program, which can be used for editing data files (its companion speed button was discussed in the previous paragraph).

File>Text Editor

Data>Spreadsheets

Data>Display

The *Data* menu includes several helpful functions as well. The *Data>Spreadsheets* command accesses UCINET's internal spreadsheet program that analysts can use for creating and editing social network matrices (see discussion of its companion speed button above). The *Data>Display* command (and its companion speed button – see discussion above) displays UCINET social

[11] UCINET has a speed button for this command in the lower right hand corner of its interface.

network files. To get a sense of how this works, issue this command, which will open a dialog box similar to that in Figure 3.2 below. Select the Krack-High-Tec.##h file and click “Open” (Note: Although you cannot see the Krack-High-Tec.##d file, it is in the folder – if it was not, you would not be able to display the data file).

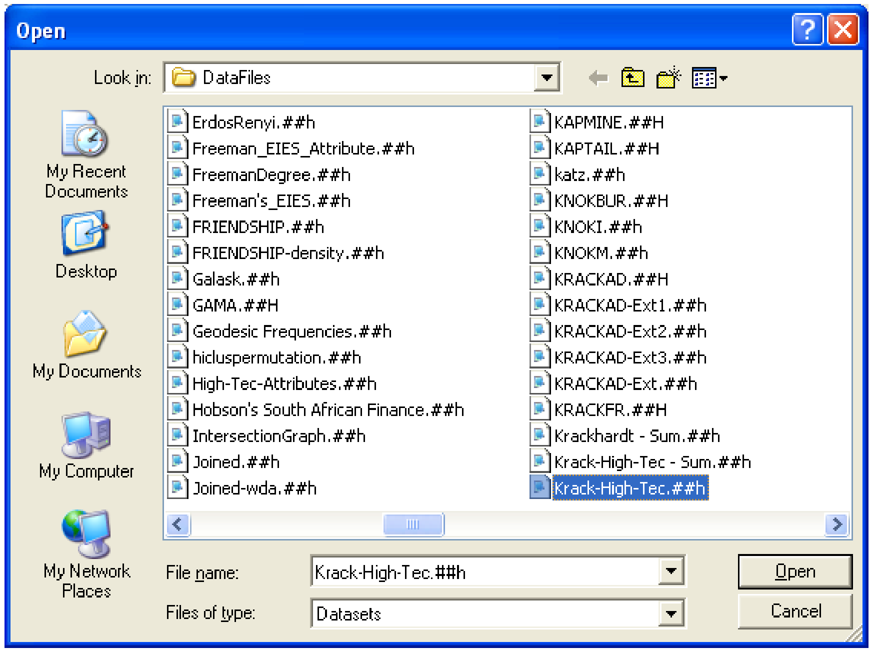
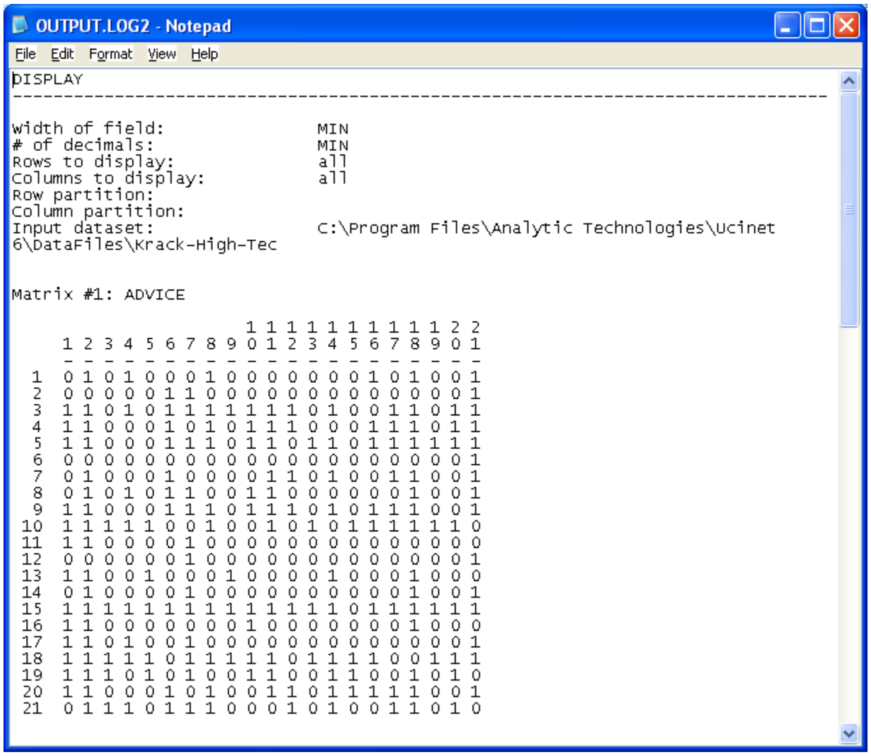


Figure 3.2: UCINET Display Dialog Box

This brings up a UCINET output log (see Figure 3.3 next page), which is generally how UCINET displays results and/or information. Note that the upper left of the output log states “Matrix #1: ADVICE”. If you scroll down the output log, you will discover that the file contains two more networks: FRIENDSHIP and REPORTS_TO. We will postpone discussing how you can (and why you might want to) “stack” networks in such a way. For now all you need to know that David Krackhardt collected data on 21 managers of a Silicon Valley company that manufactured high-tech equipment. Each manager was asked to whom they went for advice and who they considered their friends. Krackhardt also determined to whom each manager reported from company documents.

A ‘1’ recorded in the cell of the “Advice” network indicates that the manager listed on the far left seeks advice from the manager listed across the top. For instance, you can see in Figure 3.3 that manager #1 seeks advice from manager #2, but manager #2 does not seek advice from manager #1. Similarly, a ‘1’ recorded in the cell of the “Friendship” network indicates that the manager listed on the left considers the manager listed across the top to be a friend, and a

‘1’ recorded in the cell of the “Reports to” network indicates that the manager listed on the left reports to the manager listed across the top. Krackhardt also collected attribute data on the managers: age (in years), length of service or tenure (in years), level in the corporate hierarchy (1=CEO, 2 = Vice President, 3 = manager) and department (coded 1,2,3,4 with the CEO in department 0). These data, however, are stored in a different file.



```
OUTPUT.LOG2 - Notepad
File Edit Format View Help
DISPLAY
------------------------------------------------------------------------------

Width of field:               MIN
# of decimals:                MIN
Rows to display:              all
Columns to display:           all
Row partition:
Column partition:
Input dataset:                C:\Program Files\Analytic Technologies\Ucinet
6\DataFiles\Krack-High-Tec


Matrix #1: ADVICE

                         1 1 1 1 1 1 1 1 1 1 2 2
      1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
      - - - - - - - - - - - - - - - - - - - - -
  1   0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 1 0 1 0 0 1
  2   0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1
  3   1 1 0 1 0 1 1 1 1 1 1 1 0 1 0 0 1 1 0 1 1
  4   1 1 0 0 0 1 0 1 0 1 1 1 0 0 0 1 1 1 0 1 1
  5   1 1 0 0 0 1 1 1 0 1 1 0 1 1 0 1 1 1 1 1 1
  6   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1
  7   0 1 0 0 0 1 0 0 0 0 1 1 0 1 0 0 1 1 0 0 1
  8   0 1 0 1 0 1 1 0 0 1 1 0 0 0 0 0 0 1 0 0 1
  9   1 1 0 0 0 1 1 1 0 1 1 1 0 1 0 1 1 1 0 0 1
 10   1 1 1 1 1 0 0 1 0 0 1 0 1 0 1 1 1 1 1 1 0
 11   1 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 12   0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1
 13   1 1 0 0 1 0 0 0 1 0 0 0 0 1 0 0 0 1 0 0 0
 14   0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 1
 15   1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1
 16   1 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0
 17   1 1 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1
 18   1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 0 0 1 1 1
 19   1 1 1 0 1 0 1 0 0 1 1 0 0 1 1 0 0 1 0 1 0
 20   1 1 0 0 0 1 0 1 0 0 1 1 0 1 1 1 1 1 0 0 1
 21   0 1 1 1 0 1 1 1 0 0 0 1 0 1 0 0 1 1 0 1 0
```

Figure 3.3: UCINET Output Log

For now we will not explore any of the commands found under the *Transform*, *Tools* or *Network* menus. A quick glance at the Transform and Tools menus indicates that UCINET includes a number of routines for transforming and analyzing social network data. We will not utilize all of these commands in this guide, but we will use a few. The commands that will occupy most of our time are found under the *Network* menu. Here you will find routines for estimating measures of network topography (e.g., *Network> Cohesion>Density*, *Network>Cohesion>Clustering Coefficient*, *Network>Cohesion>Density*), calculating actor centrality (e.g., *Network>Centrality>Degree*, *Network> Centrality>Freeman Betweenness*, *Network>Centrality>Closeness*, *Network> Centrality>Eigenvector*), identifying subgroups (e.g., *Network>Regions> Components*, *Network>Regions>K-Core*, *Network>Subgroups>Cliques*, *Network>Subgroups>Factions*) and locating structurally equivalent actors (e.g.,

Network>Roles and Positions>Structural and *Network>Roles and Positions>Automorphic*).

Help>Help Topics

Introduction Section

Standard Datasets

Help>Hanneman Tutorial

Jumping over to the *Help* menu, the *Help>Help Topics* command leads users to its "Overview of Help" function. The "Introduction Section" link provides access to a series of help topics that covers most (but not all) of UCINET's functions (when new functions are added to UCINET, it sometimes takes awhile before they are covered). The "Overview of Help" function also provides a "Standard Datasets" link that provides users with a brief discussion of all the social network datasets that come with UCINET. Finally, the *Help>Hanneman Tutorial* command links users to the very helpful UCINET guide written by Hanneman and Riddle (2005).

The *Visualize* menu provides users with access to three visualization programs: NetDraw, Pajek and Mage. We will only use NetDraw and Pajek in this guide although it is worth mentioning that before the advent of Pajek and NetDraw, Mage was the program of choice for many social network analysts (Freeman 1999, 2000; Freeman, Webster and Kirke 1998). Mage was initially developed as a device to be used in molecular modeling (Richardson and Richardson 1992), but social network analysts found it attractive because it produces elegant three-dimensional interactive images. In Mage researchers can rotate images, turn parts of the display off and on, use the computer mouse to select and identify various parts of the network, and animate changes between different arrangements of objects.

Options >Helper Applications

Generally, you will want to leave the default settings included under the *Options* menu untouched. The one exception to this rule is the *Options>Helper Applications* command. Clicking on this command brings up the following dialog box (Figure 3.4, next page). As you can see this allows users to change the folders where the four helper applications currently incorporated by UCINET (NetDraw, Pajek, Mage and Notepad). Chances are you will never need to tell UCINET to look for NetDraw and Notepad in a different folder than the default folders set by UCINET. However, you will probably want to change where UCINET looks for Pajek (and perhaps Mage) for the simple reason that the version of Pajek (and Mage) distributed by UCINET is typically not the most recent version. You will want to download and install Pajek in its default location (i.e., C:\pajek\Pajek\Pajek.exe). So before doing anything else, click on the radio button to the right of the Pajek program location (circled in Figure 3.4 above, and change Pajek's default location to C:\pajek\Pajek\Pajek.exe. Doing this will insure that each time you open Pajek from within UCINET you will open the most recent version (assuming, of course, that you update Pajek on a regular basis).

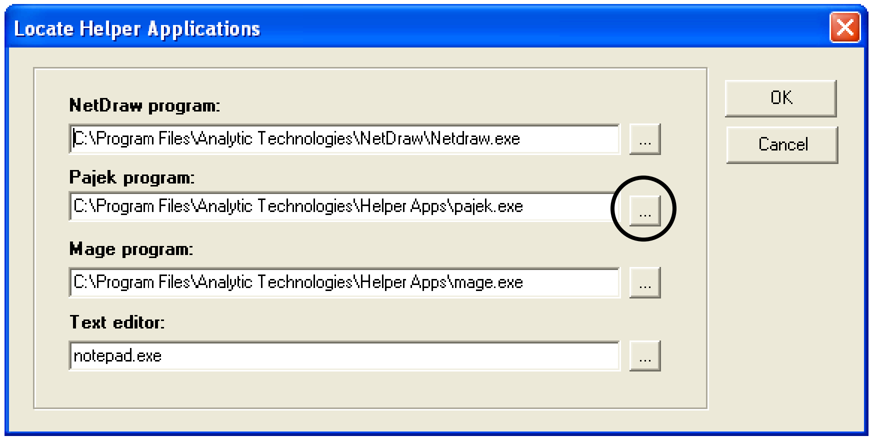


Figure 3.4: UCINET Helper Application Dialog Box

It is now time to turn our attention to NetDraw, a network visualization program that integrates nicely with UCINET. Because we will open NetDraw from within UCINET, there is no need to close UCINET at this time.

3.2 NetDraw

In recent years social network analysts have begun using a series of mathematical techniques to locate a social network's actors in such a way that the distances between them are meaningful. These algorithms use the concepts of space and distance in order to represent a network's internal structure, which they hope will reveal, among other things, which actors are "close" to one another and whether potential cleavages exist between sets of actors. NetDraw is program developed by one of UCINET's creators (Steve Borgatti) that is designed to draw networks using some of these algorithms. Network maps created by NetDraw can be rotated, flipped, resized and stored in several different formats (Huisman and van Duijn 2005:306). In addition to including various algorithms for displaying social networks in two-dimensional space, NetDraw also includes a few basic social network analysis metrics, such as calculating centrality measures and identifying subgroups.

While technically a stand-alone program, NetDraw essentially functions as an extension of UCINET: it is distributed with UCINET, it can be opened from within UCINET, and it reads UCINET files without the need for using any importing and/or exporting functions (Huisman and van Duijn 2005:306). While NetDraw's initial iterations did not handle large networks very well, more recent versions seem to do just fine. Moreover, if you save network data in .vna format, NetDraw can handle very large network files. Like UCINET, NetDraw's creators continually update the program with new procedures, routines and bug fixes.

Getting Started with NetDraw

[UCINET]
Visualize>NetDraw

To open NetDraw from UCINET, either use the *Visualize>NetDraw* command or click on the NetDraw radio button located just under the Tools menu in UCINET. This will open NetDraw's home screen, which should look something like Figure 3.5 below. Like UCINET, NetDraw has a full set of menus as well as a series of speed buttons across the top of its interface. We will consider of a few of these here.

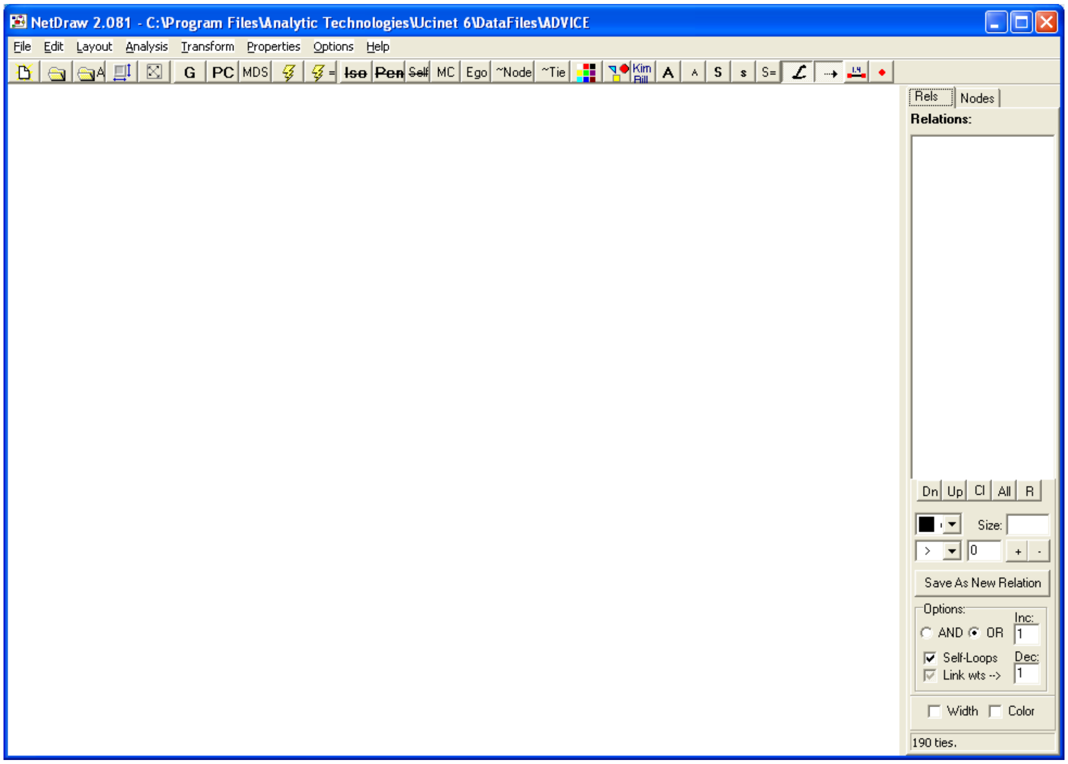


Figure 3.5: NetDraw Homepage

[NetDraw]
File>Open>Ucinet dataset
>Network

To get a sense of how NetDraw visualizes networks, open the Krackhardt dataset using NetDraw's *File>Open>Ucinet dataset> Network* command. This should yield an initial network map that looks similar to the one displayed in Figure 3.6 below:

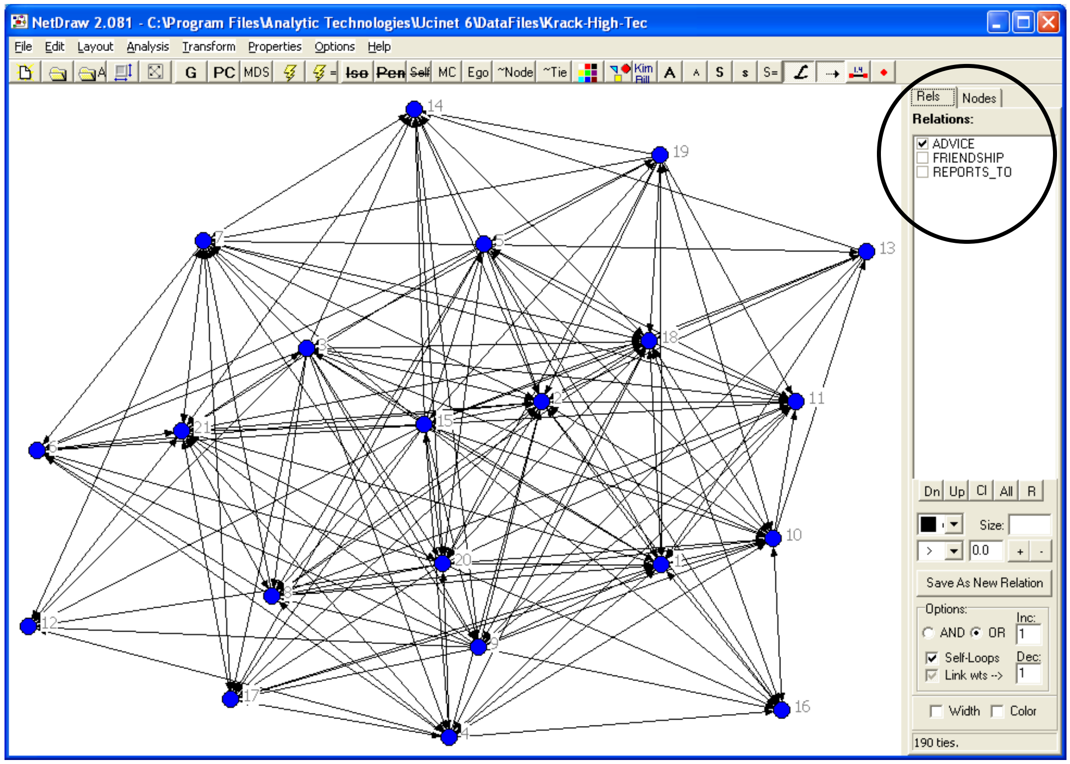


Figure 3.6: NetDraw Display of Krackhardt High Tech Data

A couple of features are worth noting before proceeding further. First, along the upper right side of NetDraw's homepage (circled in Figure 3.6) note the tab entitled "Rels" below which is located a list of the three networks (i.e., Advice, Friendship, Reports To) included in the Krackhardt file. Chances are, only the box next to the "Advice" network is checked. This indicates that only those ties are currently shown in the graph. If you check the boxes next to the "Friendship" and "Reports To" networks, NetDraw will show those ties as well. Second, NetDraw allows you to assign different colors to the various relations by using the dialog box called by the *Properties>Lines>Color>Relation* (Figure 3.7, next page). Note that NetDraw automatically assigns colors to the different types of ties. You can change these defaults by clicking on the color box. Generally, however, those chosen by NetDraw tend to provide a nice contrast between the types of ties, so it is advisable to at least begin with the default colors. You can always change them later. Note also that when two actors share more than one tie, NetDraw colors these relationships gray. You can change this default as you do the others by clicking on the color box. Once you select "Apply" NetDraw assigns the different colors to the different types of ties. There are other ways to color relations in NetDraw, but this seems to be the most stable approach (at least for now).

[NetDraw] Properties>Lines >Color>Relation

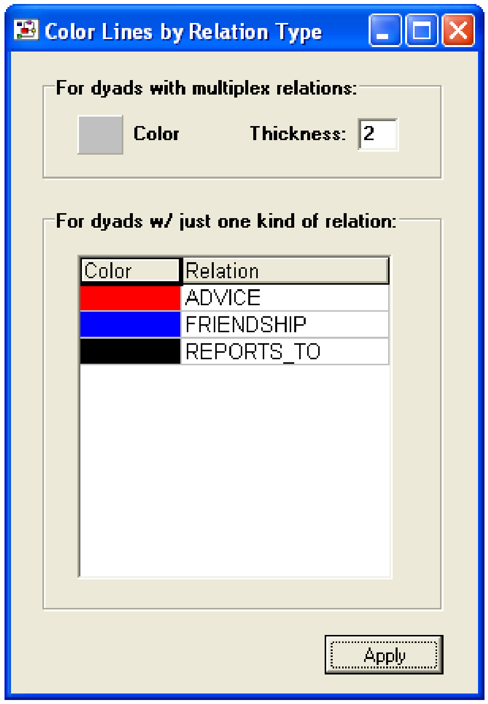


Figure 3.7: NetDraw's "Color Lines by Relation" Dialog Box

Mapping Algorithms in NetDraw

Layout>Circle

Layout>Random

Social network analysts have long used network maps to visualize social networks. A common technique in the early days of social network analysis was to construct the data around the circumference of a circle. NetDraw includes a command that creates this type of network map: *Layout>Circle*. Unfortunately, while this approach can make the structure of relations clearer, the relations between the graph's points do not reflect any specific mathematical properties. The points are arranged arbitrarily, and the distances between them are meaningless, which, depending on how they were arranged, can lead to varying interpretations of the data (McGrath, Blythe and Krackhardt 1997). NetDraw also includes an algorithm that randomly allocates a network's nodes (*Layout> Random*), but this type of layout suffers from the same weaknesses as does a circular layout: the distances between the graph's nodes mean nothing.

In recent years analysts have begun using a series of mathematical techniques to locate the points of a network in such a way that the distances between them are meaningful. Multidimensional scaling (MDS) is one such technique. It uses the concepts of space and distance to represent a network's internal structure, which, in turn, can help reveal, among other things, which actors are "close" to one another or potential cleavages between sets of actors (Wasserman and Faust 1994). The typical input to MDS is a one-mode symmetric

matrix consisting of measures of similarity or dissimilarity between pairs of actors. Output generally consists of a set of estimated distances among pairs of actors that can be then represented in one-, two-, three- or higher-dimensional space (Kruskal and Wish 1978; Wasserman and Faust 1994).

There are different types of multidimensional scaling: metric MDS and nonmetric MDS. Metric MDS takes a given matrix of proximities that measure the similarities or dissimilarities among a set of actors and calculates a set of points in k-dimensional space, such that the distances between them correspond as closely as possible to the input proximities. Metric distance differs from distance in graph theory. In graph theory, the distance between two points is measured in terms of the number of lines in the path that connects the two points. In MDS the distance between two points is the most direct route between them. "It is a distance that follows a rout 'as the crow flies', and that may be across 'open space' and need not – indeed, it normally will not – follow a graph theoretical path" (Scott 2000:148-149).

There are some limitations to using metric MDS for visualizing social networks. Many networks are binary (i.e., dichotomous) in form. They simply indicate either the presence or absence of a tie, which means that we cannot directly use such data to measure proximities. Instead, we need to first convert it into other measures, such as correlation coefficients, before calculating its metric properties. However, such data conversion can lead researchers to draw unjustifiable conclusions about the data. For example, while it is reasonable to assume that an actor with four ties is more central than one that has only two, but we cannot be certain that the former is twice as central as the latter (Scott 2000:157). Even when the data are valued, metric assumptions may be inappropriate. Imagine two actors with four ties between them and two other actors with only one. While the first set of actors is almost certainly more closely tied than the second set, again it is difficult to know *how much* more closely tied together they are (Scott 2000:157).

Non-metric MDS procedures offer a solution to this problem. Like metric MDS procedures, they use symmetrical adjacency matrices in which the cells reflect the similarities or dissimilarities among actors, but unlike metric MDS procedures, they treat the data as ordinal data, seeking "a solution in which the rank ordering of the distances is the same as the rank ordering of the original values" (Scott 2000:157). Moreover, nonmetric MDS is often preferred because it tends to provide a better "goodness-of-fit" (stress) statistic.[12]

---

[12] The lower the stress (0 = perfect fit), the better. Generally, stress levels below ".1" are considered excellent while levels above .2 are considered unacceptable.

Currently, NetDraw only includes metric MDS algorithms although plans for non-metric MDS algorithms are in the works.[13] Implementing NetDraw's metric MDS routine is relatively straightforward. Simply issue NetDraw's *Layout> Graph Theoretic layout>MDS* command (or click on its MDS speed button) and observe the change in the drawing. Nodes that are "socially" close to one another (because there is a tie between them or they are tied to a common friend) should be located close to one in the graph, while nodes that are socially distant from one another (i.e., they are not tied to one another nor do they share a tie to common tie) should be located far from one another in the graph. It is important to note that there is no single correct way to graph the data (i.e., there is not a single "solution), which is why analysts will generally want to implement an algorithm (or various algorithms) multiple times. Another metric MDS algorithm included in NetDraw is the *Layout>Graph Theoretic layout>Gower* command (or click on the G speed button). Note that all of these routines provide a slightly different network map from one another, which makes it difficult to choose which network map best fits the data. An alternative is to calculate the coordinates in UCINET and then import them into NetDraw (see Appendix 3). The advantage is that UCINET's MDS routines not only allow users to choose between metric and nonmetric algorithms, but they also provide users with a stress statistic that indicates how well the coordinates fit the data.

Layout >Graph Theoretic layout >MDS

Layout >Graph Theoretic layout >Gower

Another popular set of routines for graphing social networks are spring-embedded algorithms. Pajek, in fact, only uses spring-embedded algorithms. Spring-embedding algorithms treat the nodes as pushing and pulling on one another and seek to find an optimum solution where there is a minimum amount of stress on the springs connecting the whole set of nodes (Freeman 2000). Generally ties between nodes are treated as an attractive force (a 'spring' pulling them together) while nodes that do not share a tie are pushed apart (Moody 2001). The spring-embedding algorithm in NetDraw can be implemented using the *Layout>Graph Theoretic layout>Spring embedding* command. This brings up a dialog box (not shown) that provides you with a variety of options/criteria for using the spring embedding layout. Generally, it is a good idea to use NetDraw's defaults although varying the options tends not to change the graph's layout too dramatically. The lightening bolt speed button is another way to implement this routine, while the lightening bolt plus equals sign button implements a variation on the spring embedding routine.

Layout >Graph Theoretic layout >Spring embedding

[13] See the options found under the *Layout>Scaling/Ordination* submenu – only the Iterative metric MDS algorithm has been implemented and produces a very different solution than the other MDS routines found in NetDraw. Users can calculate metric and nonmetric MDS coordinates in UCINET and then use these coordinates in NetDraw, however. See Appendix 3 for details.

Layout
>Graph Theoretic layout
>Principal Components

NetDraw offers one additional layout algorithm – Principal Components analysis (sometimes referred to as factor analysis), which typically provides a different and often more "dramatic" graph of the network in that similar nodes tend to cluster more tightly together. NetDraw implements its Principal Components algorithm using the *Layout>Graph Theoretic layout>Principal Components* command (or the PC speed button). The logic behind principal components (factor) analysis is somewhat different than the other approaches. It is a method for combining correlated actors into a smaller number of underlying dimensions. The algorithm searches for the most highly correlated set of actors in the network; this becomes the first component. It then searches for a second set of actors that is uncorrelated with the first, which becomes the second component. Because they are uncorrelated with one another (i.e., because they are orthogonal with one another) they "can be drawn at right-angles to one another as the axes of a two-dimensional scatter diagram" (Scott 2000:154). In order for a Principal Components layout to be considered a useful map of a network, the first two dimensions (or three if you are working with a three-dimensional layout) must account for a substantial amount of the variance in the original data. If they do not, then another type of layout is in order. Unfortunately, as of now NetDraw's Principal Components routine does not tell us how much of the variance in the original it for which it accounts. Thus, analysts will need to rely somewhat on their intuition.

Working with Attributes in NetDraw

Properties>Nodes>Symbols
>Color>Attribute-based

Another nice feature of NetDraw is that it allows analysts to incorporate attribute data into its network graphs. To get a sense of this, open the attribute file (*High-Tec-Attributes.##h*) associated with the Krackhardt network data (which should still be loaded into memory), using the *File>Open>Ucinet dataset> Attribute data* command. When you click OK, a node attribute editor should appear (not shown) that is similar to a spreadsheet. Close it by clicking on the red "X" in the upper right corner. Let us first change the color of the nodes/actors to reflect the department to which they belong. To do this issue the *Properties> Nodes>Symbols>Color>Attribute-based* command, which should bring up a dialog box similar to the one in Figure 3.8. Using the drop-down box (circled in Figure 3.8 below) choose the attribute you wish to use to color the node (in this case "DEPT") and notice the change in the color of the nodes.[14]

[14] As before NetDraw automatically assigns colors for the various departments to which the managers belong.

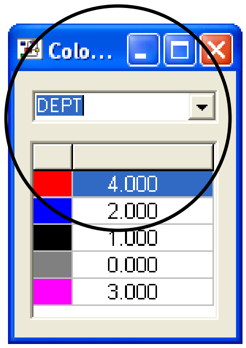


Figure 3.8: NetDraw's "Color Nodes by Color" Dialog Box

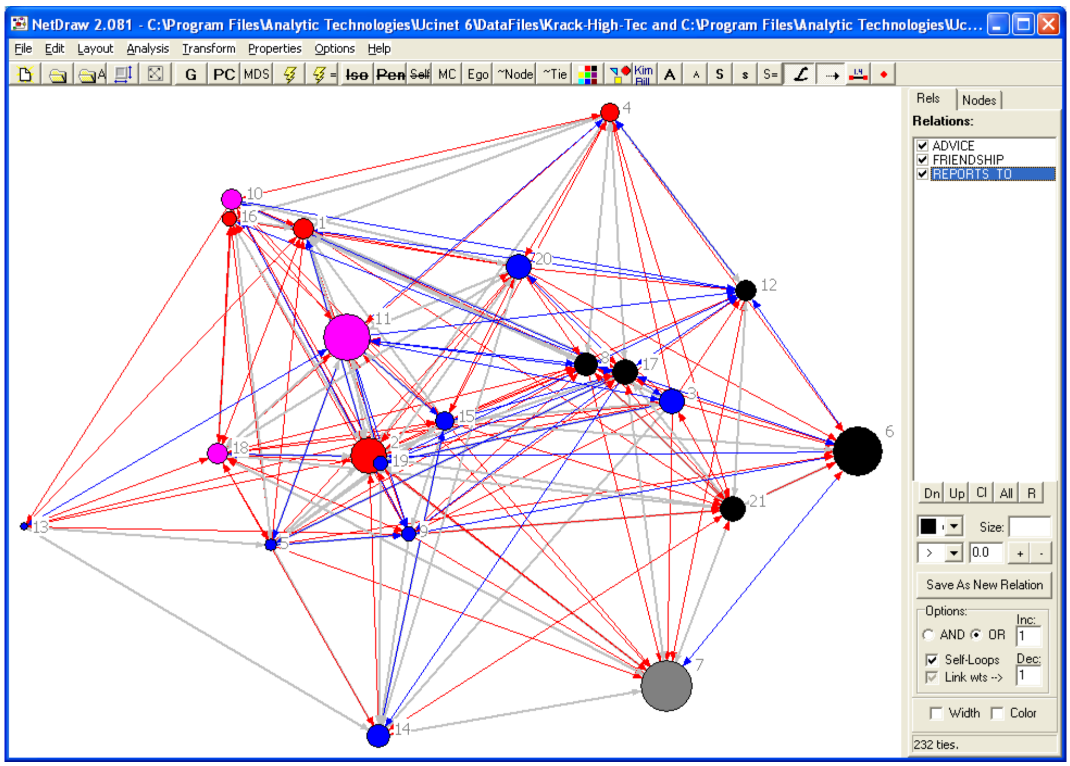


Figure 3.9: NetDraw Map of Krackhardt Data with Color and Size Reflecting Attributes

Properties>Nodes>Symbols >Size>Attribute-based

You can vary the size and shape of the nodes as well using the *Properties> Nodes>Symbols>Size>Attribute-based* and *Properties>Nodes>Symbols>Shape> Attribute-based* commands respectively. Try adjusting the *size* of the nodes based on tenure. You should get a network graph that looks somewhat similar to the one above (Figure 3.9). As you can see length of tenure does not appear to correspond with centrality in the network.

There are a number of other procedures available within NetDraw that we will consider later in the guide. Now we will turn our attention to Pajek. Before moving to a discussion of Pajek, however, we need to first save the Krackhardt data in Pajek format. To do this use NetDraw's *File>Save Data As>Pajek>Net File* command. Let us now turn our attention to Pajek.

File>Save Data As >Pajek>Net File

3.3 Pajek

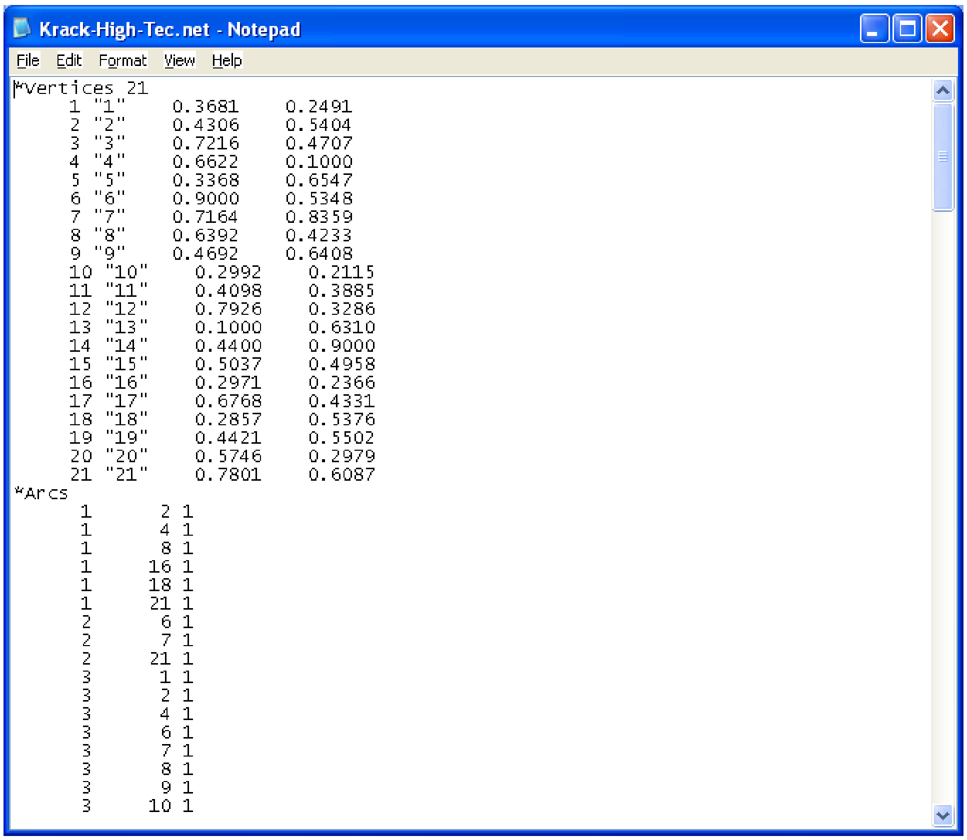
Pajek – which means "spider" in Slovenian – was created by Vladimir Batagelj and Andrej Mrvar in 1996 and is designed to handle very large datasets (Scott 2000:179-180). While it does not offer as many social network routines as does UCINET (e.g., it only computes degree, closeness and betweenness centrality), it still offers most of those that analysts use. In particular Pajek tends to include routines that lend themselves to the visualization and simplification of large social networks (Huisman and van Duijn 2005:280). Pajek runs on Windows-compatible computers, can be downloaded for free and is constantly being updated by its developers. Pajek uses six different types of data objects or structures (Huisman and van Duijn 2005:281):

- Networks (nodes/actors and ties)
- Partitions (discrete classification of nodes where each node is assigned to one *and only one* class – for example, a dark network partition might assign each actor to a specific role)
- Vectors (continuous properties of nodes – for example, an actor's age, level of education, centrality score, etc.)
- Permutations (reordering of nodes)
- Clusters (subsets of nodes)
- Hierarchies (hierarchically ordered clusters and nodes)

We will not explore all of these data objects over the course of this guide (indeed, we will utilize primarily the first three – networks, partitions and vectors), but it is helpful to at least be aware of them.

Unlike UCINET, which stores network data as a matrix, Pajek stores network data as an edge list, which is simply a list of vertices (i.e., actors/nodes) and edges/arcs (i.e., ties). Figure 3.10 (below) displays a portion of the Pajek Krackhardt High Tech network data file. The file begins by specifying the number of vertices; then each vertex is identified on a separate line by a serial number, a label (enclosed in quotation marks) and two numbers between 0 and 1, which are simply two sets of coordinates for visualizing the network data in two-

dimensional space. The list of vertices is then followed by a list of arcs. Each line identifies the number of the sending vertex, the number of the receiving vertex and the value of the tie. Thus, you can see that the first manager (vertex 1) sought advice from managers 2, 4, 8, 16, 18 and 21. One of the advantages of edge lists is that they tend to be smaller in size, and they can handle very large social networks (imagine trying to record ties in a 5,000 by 5,000 matrix).



Krack-High-Tec.net - Notepad

File Edit Format View Help

```
*Vertices 21
      1 "1"     0.3681    0.2491
      2 "2"     0.4306    0.5404
      3 "3"     0.7216    0.4707
      4 "4"     0.6622    0.1000
      5 "5"     0.3368    0.6547
      6 "6"     0.9000    0.5348
      7 "7"     0.7164    0.8359
      8 "8"     0.6392    0.4233
      9 "9"     0.4692    0.6408
     10 "10"     0.2992    0.2115
     11 "11"     0.4098    0.3885
     12 "12"     0.7926    0.3286
     13 "13"     0.1000    0.6310
     14 "14"     0.4400    0.9000
     15 "15"     0.5037    0.4958
     16 "16"     0.2971    0.2366
     17 "17"     0.6768    0.4331
     18 "18"     0.2857    0.5376
     19 "19"     0.4421    0.5502
     20 "20"     0.5746    0.2979
     21 "21"     0.7801    0.6087
*Arcs
      1      2 1
      1      4 1
      1      8 1
      1     16 1
      1     18 1
      1     21 1
      2      6 1
      2      7 1
      2     21 1
      3      1 1
      3      2 1
      3      4 1
      3      6 1
      3      7 1
      3      8 1
      3      9 1
      3     10 1
```

Figure 3.10: Pajek Net File (Edge List)

One of Pajek's nicer features is that users can load multiple networks and other data objects (e.g., partitions) into memory at the same time. This is quite helpful because, like other social network software programs, most of Pajek's routines generate new networks (or other data objects). All of these can then be stored in what Pajek calls a "project file," which means that after analyzing one or more social networks, users can save all of their work in a single file. This decreases the likelihood that users will have to later "recreate the wheel" in doing their analysis.

Pajek's primary weakness is that it only contains fewer social network procedures than UCINET (although it does include a few that UCINET does not have), and the network manipulation features are somewhat limited. That is why

many analysts sometimes use UCINET for data manipulation and then turn to Pajek for visualization purposes. To Pajek's credit, however, it now allows users to call the statistical packages "R" and "SPSS" to perform statistical procedures not available in Pajek.

Getting Started with Pajek

The Pajek main menu screen looks very different from that of UCINET and NetDraw (see Figure 3.11 below). As you can see it is organized by the type of data object or structure discussed above. And, like UCINET and NetDraw it includes a number of different menus for manipulating and analyzing data.

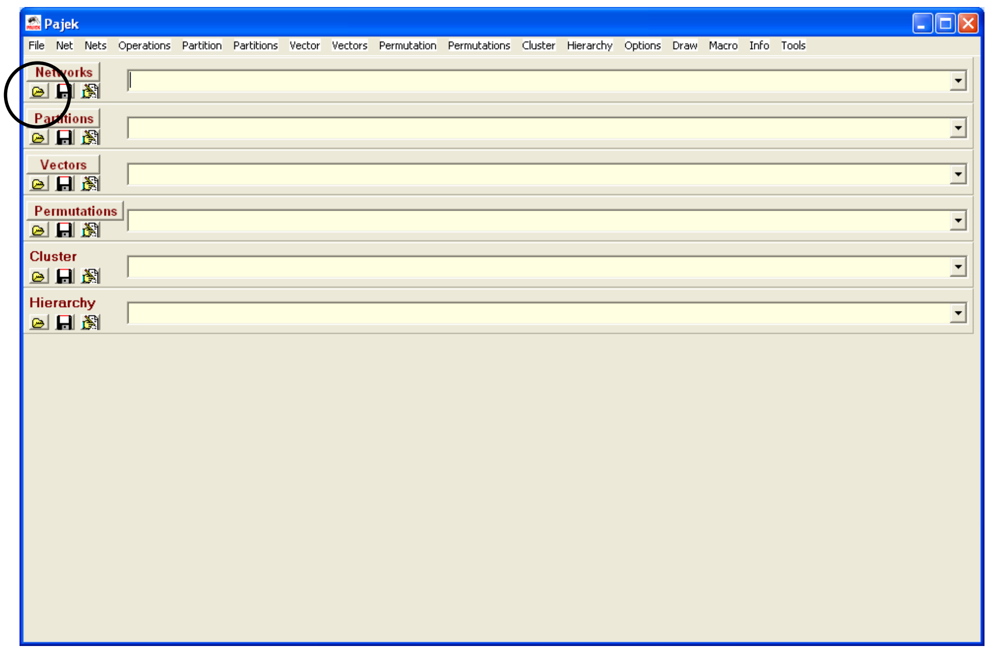


Figure 3.11: Pajek Main Menu

File>Network>Read

To open a Pajek file, use the Pajek's *File>Network>Read* command or click on the folder icon located under the "Networks" button (circled in Figure 3.11). This brings up a dialog box (not shown) that looks for different types of files. Be sure that "File Type" reads "Pajek networks (*.net)"; otherwise you will not be able to find the network data we just saved in NetDraw. Note that Pajek loads the network in the network dropdown list (not shown).[15] To view the network, use the *Pajek>Draw>Draw* command; this opens Pajek draw screen, which will look something like Figure 3.12:

Pajek>Draw>Draw

---

[15] Although it is not obvious from the dropdown list, NetDraw only saved the first network of the Krackhardt data (the advice network), which means that in Pajek we will only be able to look at the first network. There is a work around for this, but we will take that up later in the guide.

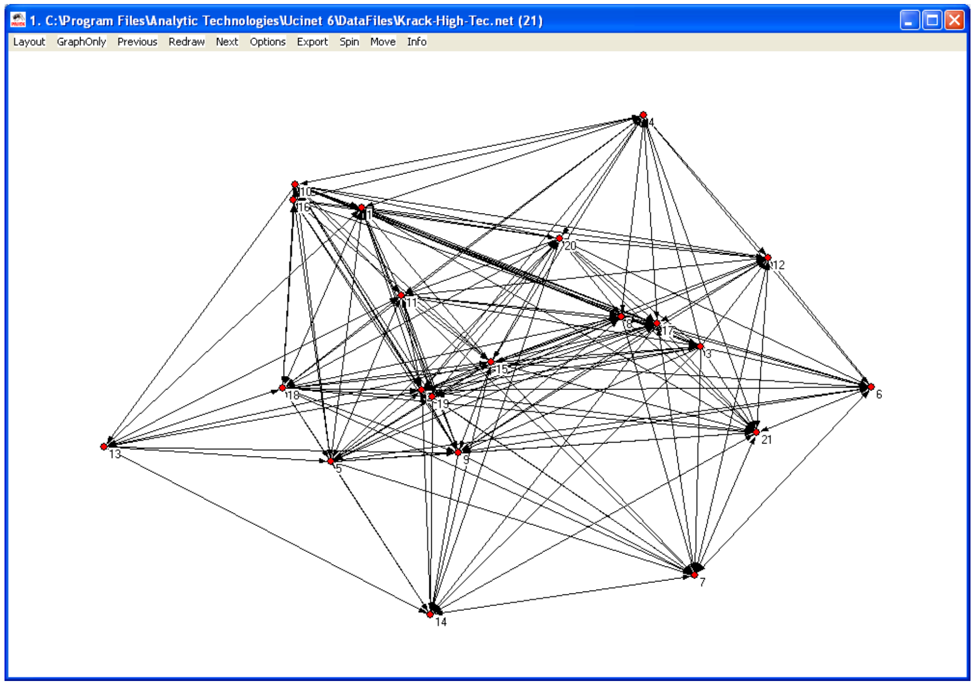


Figure 3.12: Pajek Draw Screen

[Draw Screen] Options>Color >Background

Options>Value of Lines >Similarities

Layout>Energy >Starting Positions >Given xy

Before looking at Pajek's drawing capabilities, we should choose a couple of options. To change the background use the *Options>Color>Background* command. This brings up a dialog box showing a number of colors from which to choose. Select the background of your choice. Next, you will generally want to tell Pajek that ties between actors indicate similarities or closeness between them. To do this use the *Options>Values of Lines>Similarities* command. Finally, select the *Layout>Energy>Starting Positions>Given xy* (i.e., horizontal and vertical dimensions) option; this insures that with each new drawing Pajek begins with the nodes/actors where they are (as opposed to randomly placing them before drawing the network map) and assumes that repeated drawings of networks yield a better solution

Mapping Algorithms in Pajek

As noted above Pajek only uses spring embedding algorithms for its layouts. In particular it uses two: Fruchterman Reingold and Kamada Kawai. The Kamada and Kawai (1989) algorithm is based on an assumed attraction between adjacent points (i.e., actors that are tied with one another) and an assumed repulsion between non-adjacent points (i.e., actors that are not tied to one another) and allocates points in two-dimensional space. The Fruchterman and Reingold (1991) algorithm is similar to the Kamada-Kawai algorithm, but rather than assuming attraction between adjacent points and repulsion between non-adjacent points, it

[Draw Screen]
Layout>Energy
>Fruchterman Reingold
>2D, 3D

Layout>Energy
>Kamada-Kawai>Free

attempts to simulate a system of mass particles where the vertices simulate mass points repelling each other while the edges simulate springs with attracting forces. It then tries to minimize the "energy" of this physical system. To implement Fruchterman Reingold, use Pajek's *Layout>Energy>Fruchterman Reingold>2D, 3D* command; to implement Kamada Kawai, use its *Layout>Energy>Kamada-Kawai>Free* command.

Which one should you choose? On the one hand the Kamada-Kawai algorithm works well with small, connected networks but is not recommended for networks with more than 500 vertices (de Nooy, Mrvar and Batagelj 2005:17). It draws layouts similar to nonmetric MDS and tends to be better than Fruchterman Reingold at mapping sparse networks. Unfortunately, it places isolated nodes randomly on the graph, so they can sometimes appear to be in the center of a network when in reality they are not. On the other hand, the Fruchterman Reingold algorithm is faster and works well with large networks. It also is able to distribute points in both two-dimensional and three-dimensional space. Generally, it is a good idea to first visualize the network using Fruchterman Reingold, and then (after making sure that the *Layout>Energy>Starting Positions>Given xy* option has been selected) move to using Kamada-Kawai.

[Draw Screen]

Draw>Info

Draw>Info
>Closest Vertices

Unfortunately, Pajek does not provide goodness of fit statistics for network maps, which makes it more difficult for analysts to objectively evaluate the accuracy of a network map. Pajek does, however, evaluate the esthetic properties of a network drawing with the series of commands found under the *Draw>Info* submenu. For example, some argue that the number of crossing lines in a graph should be kept to a minimum and that unconnected vertices should not be drawn too closely to one another (de Nooy, Mrvar and Batagelj 2005:18). So, if you select the *Draw>Info>Closest Vertices* command, Pajek identifies the nodes (i.e., vertices) that perform the worst on this aspect of graph drawing and gives them a different color, making them easy to identify. You can then use the mouse to drag the two vertices farther apart from one another.

Working with Attributes in Pajek

[UCINET]
Data>Export>Pajek
>Categorical Attribute

We will now look at how to work with attribute data in Pajek. In order to do this, we need to first export the Krackhardt attribute data from UCINET as Pajek partitions and vectors. In UCINET use UCINET's *Data>Export>Pajek> Categorical Attribute* command, which brings up the following dialog box:

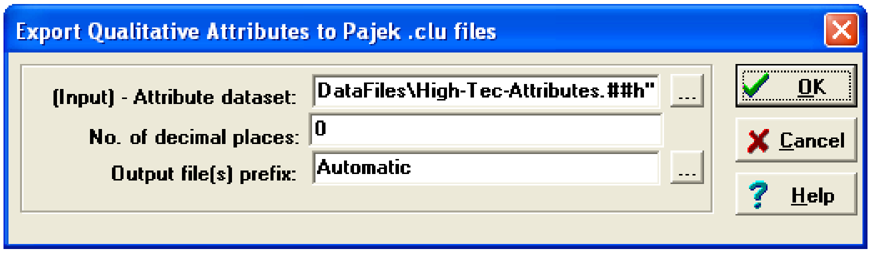


Figure 3.13: UCINET's Export Qualitative Attribute Data to Pajek Dialog Box

[UCINET] Data>Export>Pajek >Quantitative Attribute

Select the "High-Tec-Attributes.##h" file and click OK. This will create four partition files, one for each type of attribute. Technically, not all of the High Tech attributes are partitions; instead, they are vectors. Recall from the discussion above that a Pajek partition assigns each node to one and only one class. In terms of the Krackhardt data, the department attribute file fits this description. The data are not ordered (but they can be) and each manager is only belongs to one department. The other attribute data (age, level and tenure) are better thought of as vectors because they represent continuous data, so we should export these data using UCINET's *Data>Export>Pajek>Quantitative Attribute* command. This will bring up a dialog box similar to Figure 3.14. Once again select the "High-Tec-Attributes.##h" file and click OK. As before this command will create four files, one for each attribute, except that this time they will have a .vec extension.

[Pajek] File>Partition>Read

Our next task is to import the attribute files into Pajek. First, use Pajek's *File>Partition>Read* command (or click on the folder icon under the "Partition" button) in order to import the DEPT.clu file (see Figure 3.14):

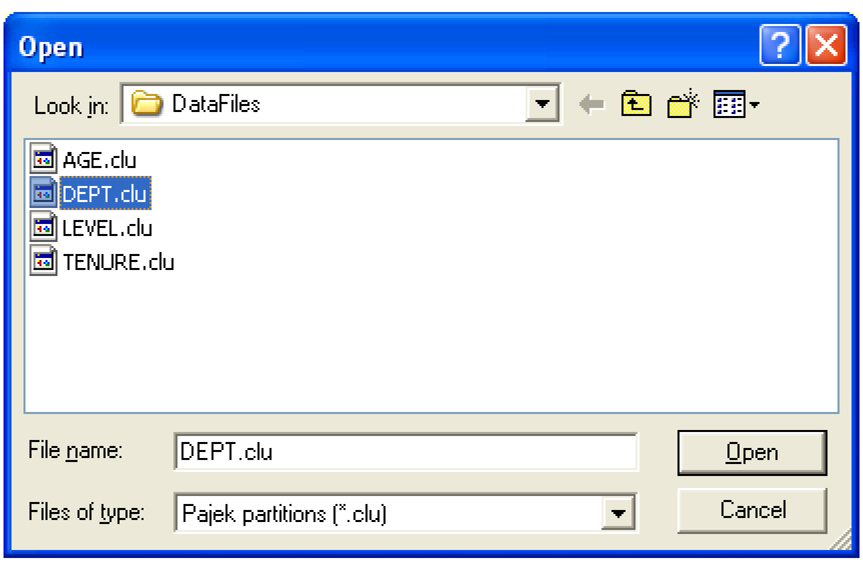


Figure 3.15: Pajek's Read Partition Dialog Box

File>Vector>Read

Next, use Pajek's *File>Vector>Read* command (or click on the folder icon under the "Vector" button) to import the age, level and tenure vectors (dialog box not shown). You have to read in each vector separately. You cannot read them in en masse. With the Krackhardt network highlighted/selected in the Network dropdown list (i.e., drop down menu), the department partition highlighted in the Partition dropdown list and the tenure vector highlighted in the Vector dropdown list (see Figure 3.16 below), select Pajek's *Draw>Draw-Partition-Vector* command and energize it using one or both of the visualization algorithms.

Draw >Draw-Partition-Vector

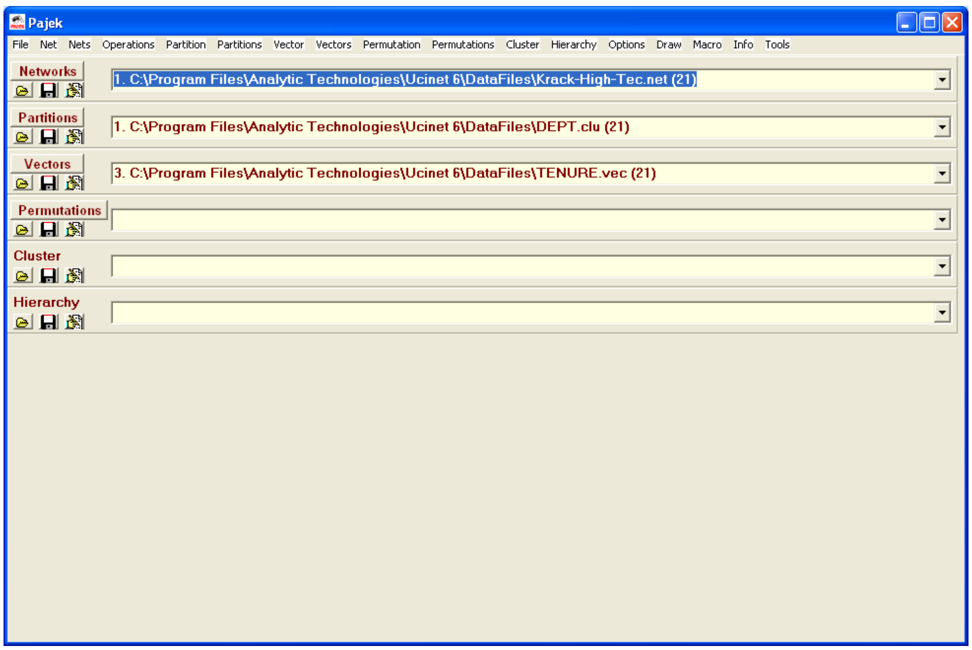


Figure 3.16: Pajek's Main Screen

[Draw Screen] Options>Size>of Vertices

This should give you a drawing similar to the one in Figure 3.17 below (next page) Here the color of the nodes indicates the department to which each manager belongs, while the size of the nodes indicates each manager's tenure. The size of the nodes in Figure 3.17 have been made somewhat larger using the *Options> Size>of Vertices* command in Pajek's Draw Screen. If you type in the number "0", Pajek automatically adjusts the size of the nodes. It is generally a good idea to let Pajek have first crack at estimating the size of each node. If they are smaller (or larger) than you like, you can always adjust them later. Here, the size has been set to "3."

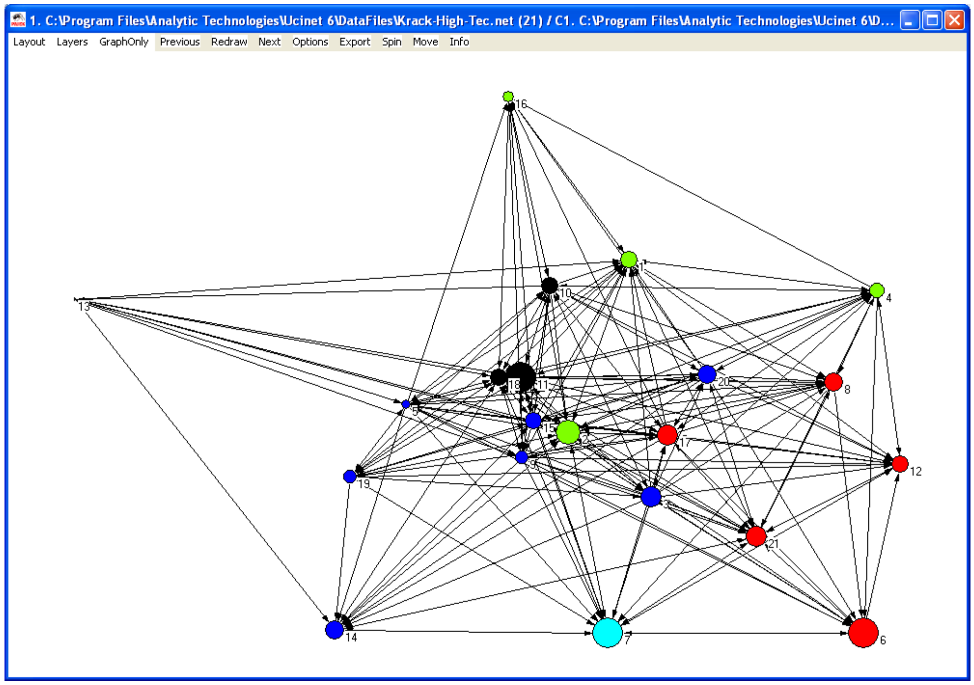


Figure 3.17: Pajek's Main Screen

3.4 Conclusion

In this chapter we have briefly examined some of the basic features of UCINET, NetDraw and Pajek. All three programs are widely used within the social network analysis community. All three have their strengths and weaknesses. Generally, analysts use UCINET for estimating metrics and manipulating data, NetDraw for the basic drawing of social networks and Pajek for more sophisticated types of drawing social networks although the lines between the three are not quite that clear cut.

CHAPTER 4

GATHERING, COLLECTING, MANIPULATING AND VISUALIZING SOCIAL NETWORK DATA: THE BASICS

Social network software packages such as UCINET, NetDraw and Pajek often come with ready-to-use data. In the real world, however, we have to collect and record our own data. Knowing how to do this is an important first step in social network analysis. We can gather and prepare social network data in a variety of ways, but most social network analysts generally record social network data in matrix form. For small datasets it is usually easiest to enter the data directly into UCINET, using its internal spreadsheet function. An advantage of entering social network data into UCINET is that we can then get it into almost every other social network analysis program since virtually all read (i.e., import) UCINET datasets. And if they do not, UCINET is usually able to export the data in a format they can read. Unfortunately, you cannot record large social networks using UCINET's internal spreadsheet because of column limitations. Moreover, it can, at times, be a bit quirky. That is why some analysts prefer to initially enter social network data into a standard spreadsheet program, such as Microsoft Excel, and then cutting and pasting it into UCINET. Not only does Excel not have the column limitations of UCINET's spreadsheet, it is also more stable and includes an "auto-complete" feature that compares the text you are typing into a cell with text already entered into the same column, which increases accuracy (e.g., consistently spelling the same name the same way each time) and input time.[16]

These considerations aside, in this chapter (and guide) we will focus our efforts on entering data using UCINET's spreadsheet program. Before turning to the nuts and bolts of how to do this, however, we need to first consider how social network analysts specify a network's boundaries, the difference between personal (ego) and complete networks, they various types of social network data, and the variety of ways that researchers collect social network data. Only then will we be ready to collect, record and manipulate actual social network data.

4.1 Boundary Specification

An important concern in social network study is which actors to include in the network and which ones not to include. Sometimes it is very clear. At other times it is not. Laumann, Marsden and Prensky (1983; 1989) note that researchers

[16] If the same word has been used before, it completes typing the entry for you.

tend to adopt various strategies for determining the boundaries of their networks. Generally, these approaches cut across two dimensions. On the one hand researchers tend to choose between realist or nominalist strategies, while on the other they tend to focus on one of three aspects of a network: actor attributes, relations, or participation in activities/events.

Realist vs. Nominalist Strategies

The realist approach is a more subjective strategy for identifying the boundaries of a network. It attempts to adopt the "vantage point of the actors themselves… [and] the network is treated as a social fact only in that it is consciously experienced… by the actors composing it" (Laumann, Marsden and Presnsky 1983:20). It assumes that the network exists as social entity for most (or perhaps all) actors of the network (Laumann, Marsden and Presnsky 1983). With this approach actors and their ties are only included to the extent that other actors consider them to be part of the network (Knoke and Yang 2007:15). Knoke and Laumann (1982) adopted this approach when selecting core U.S. energy and health national policy organizations for analysis; they only included in their analysis those organizations that energy and health policy insiders considered to be influential players in setting energy and health policies in the United States (Knoke and Yang 2007:15-16).

By contrast, the nominalist approach is a more objective approach in that rather than looking to the perceptions of network members, it imposes an *a priori* framework based on the analyst's theoretical concerns (Knoke and Yang 2007; Wasserman and Faust 1994). "For example, a researcher might be interested in studying the flow of computer messages among researchers in a scientific specialty. In such a study, the list of actors might be the collection of people who published papers on the topic in previous years. The list is constructed for the analytical purposes of the researcher, even though the scientists themselves might not perceive the list as constituting a distinctive social entity" (Wasserman and Faust 1994:32).

What should be clear is that social network analysts use both strategies in defining network boundaries. Moreover, while it is useful to draw a distinction between these two strategies, it is probably better to think of them as two poles on a continuum that runs between them where it is possible to imagine a strategy that adopts a little of both approaches in defining a network's boundary. For example, a researcher may adopt something of a nominalist approach in studying a dark network, drawing on court proceedings and newspaper accounts to initially define the network. However, she could then supplement her research through

interviews of network members, asking them to identify any other individuals and organizations that should be included in the network.

Definitional Focus: Attributes, Relations or Events

Along this dimension researchers focus on certain features of a network while leaving the remaining features free to vary (Laumann, Marsden and Presnsky 1983:22). One definitional focus that is used a lot by social network analysts is the attributes (broadly defined) of actors. We can use this focus in terms of a positional approach (i.e., where a membership test refers to the presence or absence of some attribute, such as holding a position in a formal group) or in terms of a reputational approach (e.g., one that draws on the judgments of knowledgeable informants for identifying participant actors). Examples could include the network of individuals who attend a particular faith community or the network of countries that are members of the European Union.

A relational focus for determining network boundaries focuses on a specific type of tie between actors (e.g., friendship, kinship, business, school, faith community). For example, we may be interested in studying the friendship and acquaintance ties of a particular high school. If so, then we could (theoretically) obtain a roster of the students and then ask them to identify whom they consider to be their friends.

Finally, some researchers use participation in a defining event or activity to select actors and the social relationships among them into a network. Membership in the Southern Club women network (discussed earlier) is an example of this approach (Davis, Gardner and Gardner 1941). Participation in a particular terrorist bombing might be another (Rodriguez 2005).

One thing that is important to stress is that these foci are not necessarily mutually exclusive of one another, so we will sometimes want to use them in conjunction with one another. Indeed, it may be more common for researchers to multiple foci rather than only one. For example, we may examine the friendship ties of members of a particular community of faith (rather than ties formed solely through participation on various boards and committees).

Summary

Table 4.1 (next page) combines these two dimensions into a single matrix in order to illustrate the array of possible approaches that researchers can use in order to define the boundaries of the network they are studying. What we need to make clear is that we are not limited to a single type of approach (i.e., types I through VI in the table), but rather we can adopt an approach that combines two or more foci

and lands
ultimatel
the boun
studying
somethir
destabili:

	Definitional Focus			
	Actor Attributes	Type of Relation	Event or Activity	Multiple Foci
	I	II	III	
Realist (Subjective)	• Actors included that are members of a socially defined group that group members recognize • Example: Member of a self-identified high school subgroup or clique (e.g., in-crowd, stoners, etc.)	• Focus on group members' degree or type of relation • Example: A primary (face-to-face) group or clique	• Actors' inclusion determined by participation in a series of events or activities. • Davis et al.'s study of 18 Southern women who attended 14 social events	• Combination of types I, II and/or III
	IV	V	VI	
Nominalist (Objective)	• Focus on attributes nominally defined • Example: Business elite defined as \members of Forbes 500 corporation boards	• Inclusion based on actors presence in type of relation • Example: Krackhardt's Advice and Friendship network	• Difficult to define since participation in event is self-conscious activity. • Examples: Citation networks, "invisible" college of academics where membership is defined by areas of interest (e.g., social network scholars)	Combination of types IV, V and/or VI

Note: Based on Lauman, Marsden and Presnsky. 1983. "The Boundary-Specification Problem in Network Analysis." Pp. 18-34 in *Applied Network Analysis*, edited by Ronald S. Burt and Michael Minor. Beverly Hills, CA: Sage.

Table 4.1: Boundary Specification Typology

4.2 Personal (Ego) Networks and Complete Networks

A difficulty facing social network analysts is that it is next to impossible to study very large social networks – U.S., California, Silicon Valley, the town of Palo Alto. It would be great if we could use sampling techniques because we could then generalize our findings to entire populations. Unfortunately, sampling does not work for most forms of social network analysis. Imagine what would happen if we drew a sample of 1,200 individuals and asked them who their friends were. Chances are that their friends would not be part of the sample. We simply would not have enough relational data because most of the people surveyed would not know one another. Researchers have responded to this fact in two different ways. One approach actually uses sampling but focuses only on what social network analysts refer to as ego networks, while the other approach (the more

common form approach) analyzes what social network analysts call complete social networks. We briefly consider each of these.

Ego Networks

An ego-centered approach focuses on the person surveyed – typically termed ego – and the set of contacts (i.e., alters) who have ties to the person and measurements on the ties among these alters. Each person surveyed is generally asked for a set of contacts (Burt 1984, 1985), using questions such as "Looking back over the last six months, who are the people with whom you discussed matters important to you?" After providing a list of contacts, they are then asked about the ties (if any) between their contacts (e.g., do they know one another, do they attend the same church, are they friends, and so on). Needless to say, ties between an ego and his or her alters with different egos and their alters are not (and generally cannot be) recorded. This yields a data structure similar to that displayed in Figure 4.1 below. As one can see, only those ties within the ego-network of the people sampled (i.e., Doug and Nancy) are recorded, while ties between ego-networks are not.

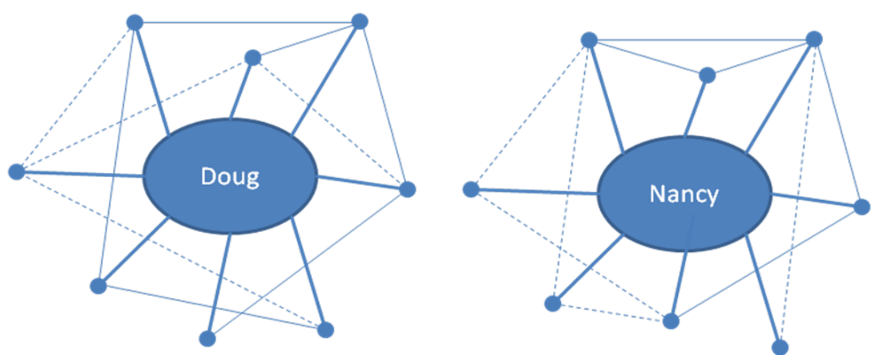


Figure 4.1: Hypothetical Ego Network

A common use of ego-network data is to estimate the size of peoples' core networks to see if it has changed over time or is correlated with certain types of behavior. For example, Miller McPherson, Lynn Smith-Lovin and Matthew Brashears (2006) found that from 1985 to 2004 the average size of individuals' core discussion networks dropped from 2.94 to 2.08, while the modal size dropped from 3.00 to 0.00. In fact, almost one quarter of the population now reports that they do not discuss important matters with anyone!

Complete Networks

A more common approach to social network analysis collects relational data on an entire network (assuming that we know the boundaries of that network—see section 4.1 above). Most social network methodologies are built on the assumption that the network being studied is a complete network that not only includes all relevant actors but also all relevant ties between actors. Because the complete network approach places limits on the size of the networks that can be studied, social network analysts focus primarily on case studies, which is what we will do in this guide.

4.3 Types of Social Network Data

Social network analysts work with three types of data: one-mode social network data (symmetric and asymmetric), two-mode social network data and attribute data. We discussed the nature of attribute data earlier (Chapter 2), so here we limit our discussion to one and two-mode network data.

Symmetric One-mode Networks

One-mode networks consist of a single set of actors, which can be people, groups, organizations, corporations, nation-states, etc. The ties between actors can be friendship or kinship ties, material transactions such as business transactions, the import or export of goods, communication networks involving the sending or receiving of messages, etc. An example of a one-mode network, which we briefly discussed in the first chapter, is Padgett and Ansell's (1993) Florentine Families Network. Padgett and Ansell collected nine types of relational data on 92 prominent 15th century Florentine families in order to explain Cosimo de' Medici's rise to power. Included with UCINET 6.0 is a subset of this data that delineates the business and marriage ties between 16 of the 92 families. A marital tie was determined to exist if a member of one family married a member of another family, while a business tie was determined to exist if a member of one family granted credits, made a loan, or entered into a joint partnership with a member of another family (Wasserman and Faust 1994). Figure 4.2 (next page) presents a UCINET screen shot of the Padgett and Ansell's marriage data.

As noted earlier, one-mode networks always result in square matrices because each actor (in this case, each family) appears as both a row and a column. For example, the Acciaiuol family's ties are recorded in both the first row and first column, the Albizzi family's ties are recorded in the second row and second column, the Barbadori's in the third row and column, and so on. In this case the

ties are considered *dichotomous* because they only take the values of “0” or “1” with “1” indicating the presence of a marriage tie and “0” indicating the absence of one.[17] From this matrix you can see that the Medici family had ties to six families – Acciaiuol, Albizzi, Barbadori, Ridolfi, Salviati and Tornabuon – while, the Strozzi family had ties to four – Bishcheri, Castellan, Peruzzi, Salviati.

		1 A	2 A	3 B	4 B	5 C	6 G	7 G	8 L	9 M	10 P	11 P	12 P	13 R	14 S	15 S	16 T
1	ACCIAIUOL	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
2	ALBIZZI	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0
3	BARBADORI	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
4	BISCHERI	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0
5	CASTELLAN	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0
6	GINORI	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	GUADAGNI	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	1
8	LAMBERTES	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
9	MEDICI	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	1
10	PAZZI	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
11	PERUZZI	0	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0
12	PUCCI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	RIDOLFI	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	1
14	SALVIATI	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
15	STROZZI	0	0	0	1	1	0	0	0	0	0	1	0	1	0	0	0
16	TORNABUON	0	0	0	0	0	0	1	0	1	0	0	0	1	0	0	0

Figure 4.2: Subset of Padgett and Ansell’s Marriage Data

Asymmetric One-mode Networks

Ties between actors are not always reciprocal. Take, for example, the social network data collected by David Krackhardt (1987) on 21 managers in a Silicon Valley high-technology company that had approximately 100 employees. Krackhardt asked each manager to whom they went to for advice and whom they considered a friend. He also determined from company documents to whom each manager reported. Figure 4.3 presents the advice network data (in matrix form). This matrix indicates that while virtually every manager seeks advice from Managers #2 & #21, Managers #2 and #21 do not always reciprocate. For example, while Manager #1 seeks advice from Manager #21, Manager #21 does not seek advice from Manager #1 in return. By contrast, Manager #15 is not too popular in terms of giving advice although he is not shy in asking for it himself. While he seeks advice from every other manager, only four (#10, 18, 19 & 20) seek his in return.

[17] Cell values can also be “valued” indicating some sort of numerical relationship between two actors. For example, a cell may indicate the amount of imports between two countries.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
1	0	1	0	1	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	1
2	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
3	1	1	0	1	0	1	1	1	1	1	1	1	0	1	0	0	1	1	0	1	1
4	1	1	0	0	0	1	0	1	0	1	1	1	0	0	0	1	1	1	0	1	1
5	1	1	0	0	0	1	1	1	0	1	1	0	1	1	0	1	1	1	1	1	1
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
7	0	1	0	0	0	1	0	0	0	0	1	1	0	1	0	0	1	1	0	0	1
8	0	1	0	1	0	1	1	0	0	1	1	0	0	0	0	0	0	1	0	0	1
9	1	1	0	0	0	1	1	1	0	1	1	1	0	1	0	1	1	1	0	0	1
10	1	1	1	1	1	0	0	1	0	0	1	0	1	0	1	1	1	1	1	1	0
11	1	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
13	1	1	0	0	1	0	0	0	1	0	0	0	0	1	0	0	0	1	0	0	0
14	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	1
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
16	1	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0
17	1	1	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1
18	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	0	0	1	1	1
19	1	1	1	0	1	0	1	0	0	1	1	0	0	1	1	0	0	1	0	1	0
20	1	1	0	0	0	1	0	1	0	0	1	1	0	1	1	1	1	1	0	0	1
21	0	1	1	1	0	1	1	1	0	0	0	1	0	1	0	0	1	1	0	1	0

Figure 4.3: Krackhardt High-Technology Manager Advice Network Data

Two-mode networks

Two-mode networks differ from one-mode networks in that rather than consisting of a single set of actors, they either consist of two sets of different actors, or one set of actors and one set of events or affiliations. Researchers often refer to them as *affiliation networks*, but they sometimes call them *membership networks*, *dual networks* and *hypernetworks*. Examples of two-mode networks include membership in various organizations, attendance at particular events, employees at a particular company and so on. An example of a two-mode network is Davis's Southern Club Women (Breiger 1974; Davis, Gardner and Gardner 1941). Davis and his colleagues recorded the observed attendance of 18 Southern women at 14 social events. The women are listed by row, while the events are listed by column. As Figure 4.3 indicates Evelyn attended eight events (#'s 1, 2, 3, 4, 5, 6, 8 & 9), while Olivia and Flora only attended two (#'s 9 & 11).

A key assumption underlying the use of two-mode networks by social network analysts is that membership in an organization or participation in an event is a source of social ties. Why? Because people who join or participate in a common organization and/or event often share similar tasks and/or interests, and they are much more likely to interact with one another than two randomly selected people. That said, we need to be careful when using two-mode data. For instance, just because two people participate in a common event or are members of the same faith community does not necessarily mean that a tie exists between them.

		1 E	2 E	3 E	4 E	5 E	6 E	7 E	8 E	9 E	10 E	11 E	12 E	13 E	14 E
1	EVELYN	1	1	1	1	1	1	0	1	1	0	0	0	0	0
2	LAURA	1	1	1	0	1	1	1	1	0	0	0	0	0	0
3	THERESA	0	1	1	1	1	1	1	1	1	0	0	0	0	0
4	BRENDA	1	0	1	1	1	1	1	1	0	0	0	0	0	0
5	CHARLOTTE	0	0	1	1	1	0	1	0	0	0	0	0	0	0
6	FRANCES	0	0	1	0	1	1	0	1	0	0	0	0	0	0
7	ELEANOR	0	0	0	0	1	1	1	1	0	0	0	0	0	0
8	PEARL	0	0	0	0	0	1	0	1	1	0	0	0	0	0
9	RUTH	0	0	0	0	1	0	1	1	1	0	0	0	0	0
10	VERNE	0	0	0	0	0	0	1	1	1	0	0	1	0	0
11	MYRNA	0	0	0	0	0	0	0	1	1	1	0	1	0	0
12	KATHERINE	0	0	0	0	0	0	0	1	1	1	0	1	1	1
13	SYLVIA	0	0	0	0	0	0	1	1	1	1	0	1	1	1
14	NORA	0	0	0	0	0	1	1	0	1	1	1	1	1	1
15	HELEN	0	0	0	0	0	0	1	1	0	1	1	1	0	0
16	DOROTHY	0	0	0	0	0	0	0	1	1	0	0	0	0	0
17	OLIVIA	0	0	0	0	0	0	0	0	1	0	1	0	0	0
18	FLORA	0	0	0	0	0	0	0	0	1	0	1	0	0	0

Figure 4.3: Davis's Southern Club Women

4.4 Collecting Social Network Data

Social network analysts collect social network data in a variety of ways. The most common are questionnaires, interviews, direct observation and written records (Wasserman and Faust 1994:45-54). While not all of these are useful for collecting data on dark networks, it is still useful to briefly look at these various approaches for collecting social network data.

Questionnaires

Questionnaires are a common method for collecting social network data, especially when actors are people. They contain questions such about whom people consider to be their friends, to whom do they go to for advice, with whom do they regularly communicate (e.g., talk face-to-face, email, telephone), and so on. Such data can be recorded either symmetrically or asymmetrically. Say, for instance, actor "A" considers actor "B" to be a friend, but "B" does not consider "A" to be a friend. In such a case researchers can either record the data as Krackhardt did (i.e., asymmetrically) by placing a "1" in the "A-B" cell of the matrix but a "0" in the "B-A" cell of the matrix, or they can record it symmetrically by placing a "0" in both cells under the assumption that a friendship tie only exists if both actors indicate that they consider the other to be a friend. Analysts use different questionnaire formats for collecting social network data. These fall under three broad categories: (1) Roster vs. Free Recall, (2) Free

vs. Fixed Choice, and (3) Ratings vs. Complete Rankings (Wasserman and Faust 1994:45).

Roster vs. Free Recall

Sometimes analysts present each respondent filling out a questionnaire with a complete roster of the actors in the network or allow the respondents to generate a list of names. Rosters can only be used when researchers know the members in the network prior gathering data (Wasserman and Faust 1994:46), but this raises the network boundary issues discussed earlier: how do researchers know *a priori* which actors belong in the network and which ones do not? When working with a self-contained organization (e.g., a small high technology start-up) this is sometimes relatively obvious (at least for the purposes of the study), but it is not always so clear-cut. In the latter case, it is usually advisable to use the free recall approach.

Free vs. Fixed Choice

In some network designs analysts tell respondents how many other actors they are to nominate on a questionnaire (e.g., " Name five people with whom you have regular contact"); at other times they are not presented with any such constraints as to how many nominations they can make (e.g., "Name everyone with whom you have regular contact"). The former (fixed choice) can underestimate the size or density of a network and thus misleading results.

Ratings vs. Complete Rankings

Finally, sometimes analysts ask each respondent to rate or rank the ties in terms of strength between all actors in the network (Wasserman and Faust 1994:48). Ratings can be dichotomous (e.g., ties are either present or absent) or they can be valued (e.g., respondents choose one of a few possible categories for the strength of each tie). Rankings differ in that each actor is asked to rank their ties to every other actor in the network. This latter approach becomes increasingly difficult as the size of the network increases.

Interviews

Social network analysts sometimes use interviews (either face-to-face or over the phone) when the use of questionnaires is not possible (Wasserman and Faust 1994:48). Interviews with captured members of terrorist groups may prove useful for mapping dark networks, but these should probably be supplemented with other methods (e.g., written records).

Direct Observation

Another way to record data is to have an observer record all interactions that take place among actors in the network (Wasserman and Faust 1994:49). Dan McFarland (2004) used this approach to record student interaction patterns at two different high schools. An obvious drawback to this approach is that in some situations interactions can be so numerous and occur so closely together that it becomes next to impossible to record all interactions. Still, analysts of dark networks might find this approach useful when recording affiliation (i.e., two-mode) network data. For example, they can record which members of a particular dark network visit specific sites or attend specific events in connection with their participation with the network.

Written Records

Written records can be valuable sources of relational data. Emails, memos, phone calls (if available), historical marriage records and court proceedings are just a few examples of sources from which one can determine ties between individual actors. Sageman (2004b; 2008:26-27), for instance, drew on captured documents, trial transcripts, intercepted conversations, legal documents and testimony notes in order to determine some of the ties among members of the global salafi jihad, and throughout this guide we will draw on narrative about Noordin's terrorist network also drew on court records (International Crisis Group 2006). At the corporate level written records indicating joint ventures, interlocking directorates (i.e., where the same individual sits on the boards of two different companies), and membership in the same trade association may indicate ties, while records indicating the trade manufactured goods or the exchange of diplomats may indicate ties between countries.

Other Approaches

These are not the only approaches to collecting social network data. They are simply the most common. Other forms of data collection include cognitive social structures, experiments, dairies and small worlds (Wasserman and Faust 1994:51-54). When collecting cognitive social structure data researchers ask respondents for their *perception* of other actors' network ties (e.g., "Who is friends with whom?") (Krackhardt 1987). Social network analysts sometimes use experiments to observe the behavior of a set of actors in experimentally controlled environments (Bavelas 1950; Emerson 1962). Diaries are used by social network analysts to ask respondents to keep a continuous record of people with whom they

interact (Wasserman and Faust 1994:54). Finally, researchers will use variations on small work network design (Milgram 1967; Travers and Milgram 1969) to estimate how many actors a respondent is removed from a randomly chosen target (Watts, Dodds and Newman 2003).

4.5 Recording Social Network Data

To illustrate how to record social network data, we begin with relatively simple one-mode social networks (both symmetric and asymmetric). We then examine how to record larger and more complex social network data: namely, data gleaned from Noordin Top's Terrorist Network (International Crisis Group 2006). Next we examine two-mode social network data, beginning (as before) with a simple two-mode network (Davis's Southern Women) before looking at two-mode example of Noordin's network. In this section we also examine how to transform two-mode networks into one-mode networks, how to export both so that they can be read into Pajek and how to record attribute data.

One-Mode Social Network Data

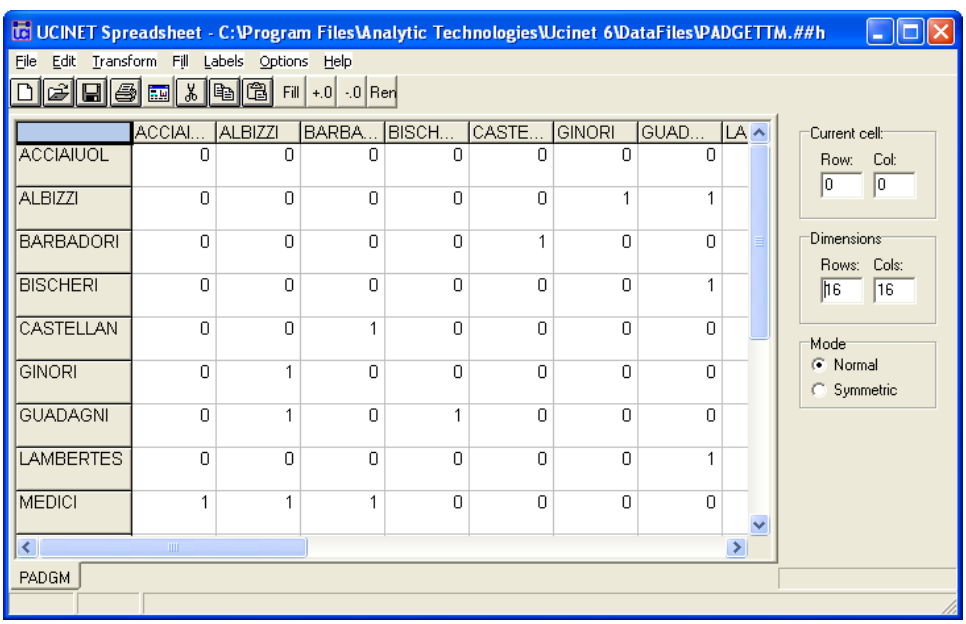


Figure 4.5: Padgett and Ansell's Marriage Network

Data >Spreadsheet Editor

One-mode social network data can easily be recorded in matrix form using UCINET's internal spreadsheet editor. For example, if we wanted to enter Padgett Marriage Data illustrated above, we would first open the spreadsheet editor (found under UCINET's Data menu) and enter "1's" wherever a tie between families

exists (see Figure 4.5 above). We can either enter the "0's" as we go or only enter cell values that are greater or less than "0". If we choose the latter, after you have entered all of the values, we need to first enter the "Dimensions" of the matrix (i.e., the number of rows and columns) on the right side of the Spreadsheet editor and then click on the "Fill" button, which will fill all empty cells with "0's".

Data >Spreadsheet Editor >Fill>Blanks w/0s

Noordin Top's Terror Network

Let us now turn our attention to the terrorist network of Noordin Mohammed Top, the group believed to behind the 2003 Marriott Hotel and the 2004 Australian Embassy bombings in Jakarta, and the 2005 Bali bombings (International Crisis Group 2006). While a number of individuals have ties to Noordin and his terrorist group, for our purposes we use the friendship network of the 79 individuals listed in appendix of the International Crisis Group's (2006) account of Noordin's operations. A friendship tie is defined as a close attachment built on affection or esteem between two people, not including casual meetings and/or school ties. Figure 4.6 presents a portion of Noordin's Friendship network after it has been entered into UCINET's spreadsheet function. Note that the form is no different than that of the smaller datasets. Each actor appears as both a row and a column. Needless to say, after entering social network data, we need to save it, preferably with a file name that is easily identified.

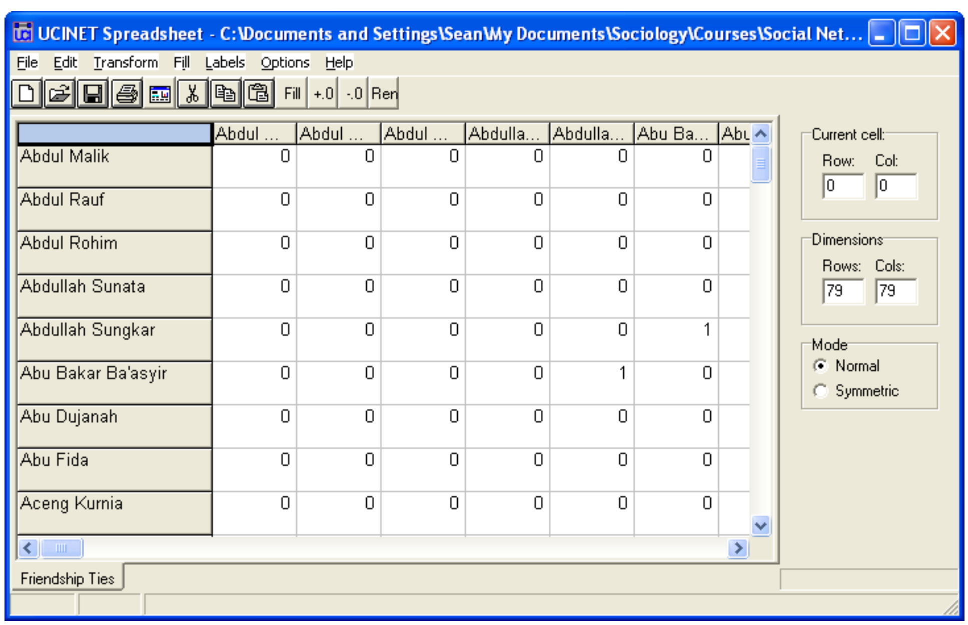


Figure 4.6: Noordin Top's Friendship Network

Exporting One-Mode Data to Pajek

There are multiple methods for exporting one-mode social network data to Pajek. The easiest, assuming that you already have the data in UCINET format, is to simply export the data from UCINET in Pajek form.[18] Under the “Data” menu, select the *Export>Pajek>Network* command (note that UCINET provides a choice of exporting the data in a number of formats: DL, Krackplot, Mage, Pajek, Metis, Raw, and Excel). This brings a dialog box (Figure 4.7, next page). UCINET provides a series of options although most of the time you will want to accept UCINET’s default settings. A good rule of thumb is that if you do not know what the option is asking, then accept UCINET’s default (Note: you can click on the dialog box’s Help button for a more detailed discussion of each of the options).

Data>Export >Pajek>Network

The one exception to the rule here concerns the final option, which allows you to launch Pajek from within UCINET once the data are exported. You only want to use this option if (1) you do not already have Pajek open and (2) you have set UCINET’s options so that it will open the most recent version of Pajek. Otherwise, it will launch the version of Pajek that comes with UCINET, which is often somewhat dated. If you choose yes, another dialog box will appear asking you (again) whether you want to launch Pajek. If all goes well, UCINET launches Pajek when you click the “OK” button. If not, you can open Pajek and import the newly created Pajek network file manually.

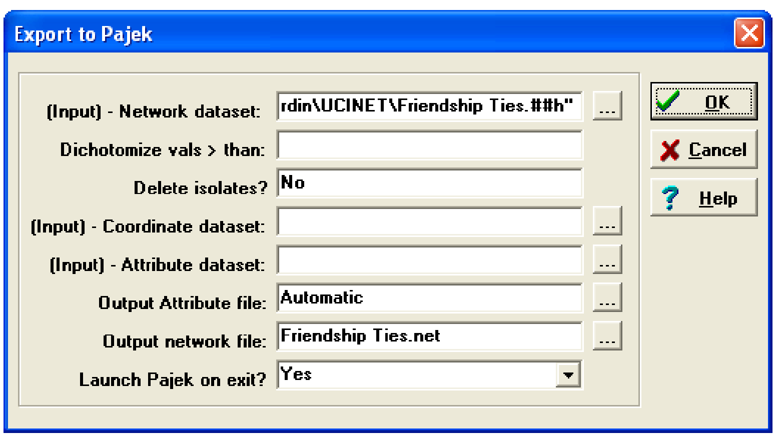


Figure 4.7: UCINET Export to Pajek Dialog Box

Because Pajek now allows users to read in multiple (i.e., stacked) networks (see discussion in Section 4.6 below), analysts may want to use UCINET’s *Data>Export>DL* command when exporting multiple networks. We have to use

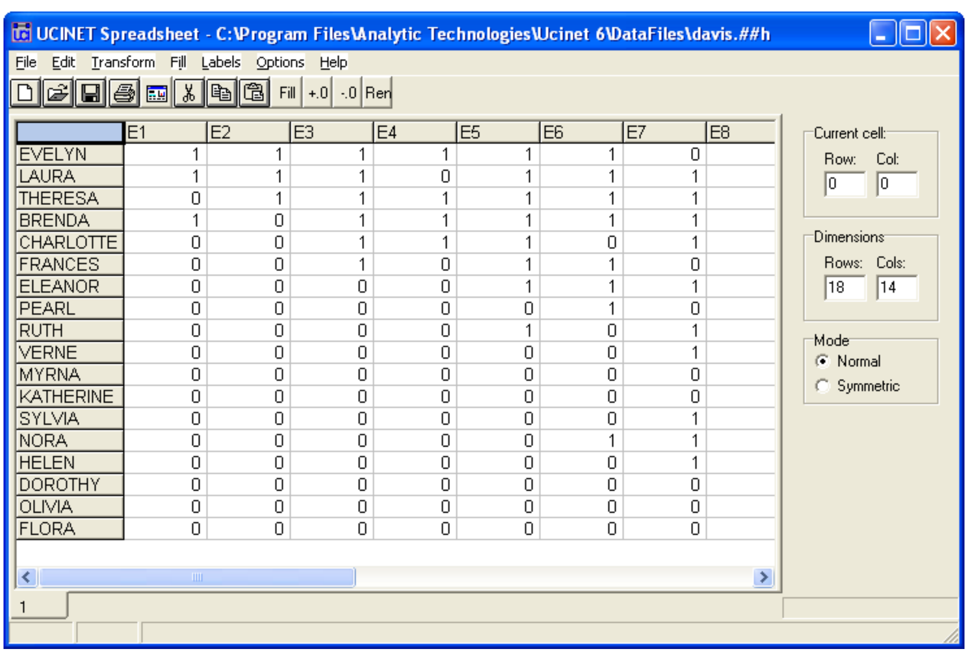
Data>Export>DL

[18] For now, UCINET 6.0 does not export two-mode social network data in Pajek format. Different techniques need to be utilized (see below).

this feature when exporting two-mode data from UCINET to Pajek, so we delay discussing this command until then. We also illustrate how to export multiple networks in Section 4.6.

Two-Mode Social Network Data

We enter two-mode data into UCINET's spreadsheet function just as we did with one-mode data, but the form of data entry differs somewhat. Taking Davis's Southern Club Women as an example, the names of the women appear in rows, while the columns list the various events that the women attended (Figure 4.8, next page). This could just as easily be reversed where the names of the events appear in rows, and the names of the women appear in the columns. However, social network analysts generally work with an implicit "left to right" logic; thus, since the women are more "logically" seen as attending various events (rather than the events attracting the women – although this is true as well), the women appear first (in rows) while the events appear second (in columns). The same logic appears when working with actors' membership in various institutions. For instance, if a series of actors were members of a particular set of church-related groups, generally the actors would be listed in the rows and the groups listed in the columns. Here, we can see that Evelyn attended events E1 through E6 (she also attended events E8 and E9, but we cannot see the entire matrix in this screen shot), while Laura attend events E1-E3 and E5-E7.



	E1	E2	E3	E4	E5	E6	E7	E8
EVELYN	1	1	1	1	1	1	0	
LAURA	1	1	1	0	1	1	1	
THERESA	0	1	1	1	1	1	1	
BRENDA	1	0	1	1	1	1	1	
CHARLOTTE	0	0	1	1	1	0	1	
FRANCES	0	0	1	0	1	1	0	
ELEANOR	0	0	0	0	1	1	1	
PEARL	0	0	0	0	0	1	0	
RUTH	0	0	0	0	1	0	1	
VERNE	0	0	0	0	0	0	1	
MYRNA	0	0	0	0	0	0	0	
KATHERINE	0	0	0	0	0	0	0	
SYLVIA	0	0	0	0	0	0	1	
NORA	0	0	0	0	0	1	1	
HELEN	0	0	0	0	0	0	1	
DOROTHY	0	0	0	0	0	0	0	
OLIVIA	0	0	0	0	0	0	0	
FLORA	0	0	0	0	0	0	0	

Figure 4.8: Davis's Southern Club Women

Noordin Top's Terror Network

The Noordin data include several affiliation networks: school ties, religious ties, operational (bombing) ties, training ties, etc. While these data are larger than Davis's Southern Club women, we record them in exactly the same way. Figure 4.9 (next page) displays a slice of Noordin Top's Operational Network. The names of Noordin's network members are listed in rows, while the names of four operations (Australian Embassy, Bali I, Bali II and the Marriott Hotel) are listed in columns. If a member of Noordin's network participated in a particular bombing, then a "1" appears in that cell. If he did not, then a "0" appears. As you can see, Abdul Rauf participated in the Bali I bombing while Achmad Hasan participated in the Australian Embassy bombing. After we finish entering the data, we need to save it, preferably with a file name that is easily identified.

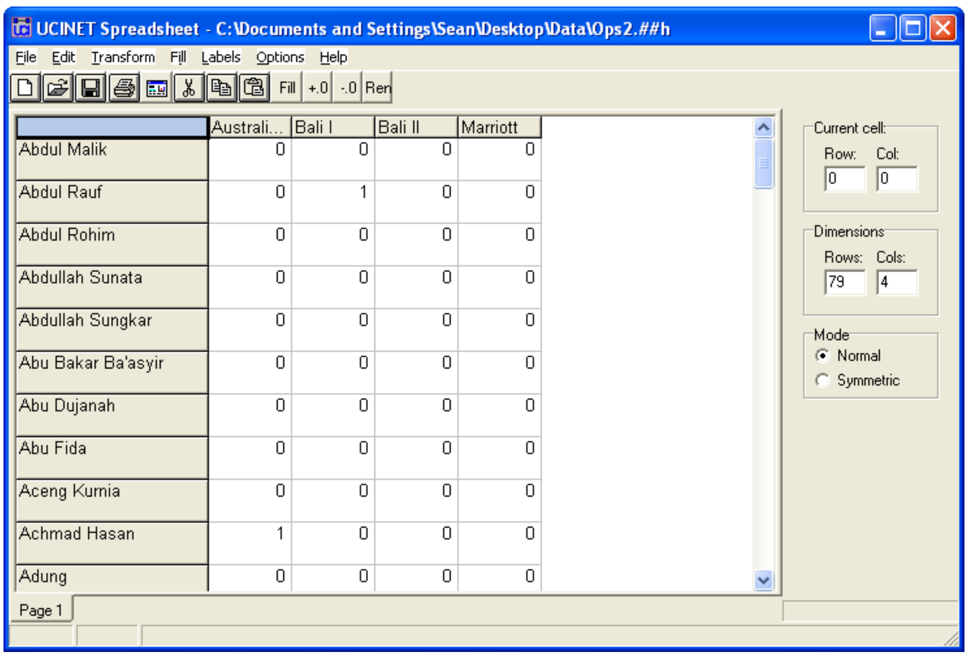


Figure 4.9: Noordin Top's Operational Network

Exporting Two-mode Data to Pajek

[UCINET] Data>Export>DL

Unfortunately, UCINET 6.0 currently does not export two-mode networks in Pajek format, so we have to export it differently than we did with one-mode networks. In UCINET choose the *Data>Export>DL* command; this brings up a dialog box (see Figure 4.10, next page). Again, several options are offered although you will generally want to accept UCINET's defaults. In this case accept all of UCINET's defaults except the last: manually change the extension of the

output dataset to ".dat" (rather than ".txt") because Pajek looks for DL files with .dat extensions not .txt ones.

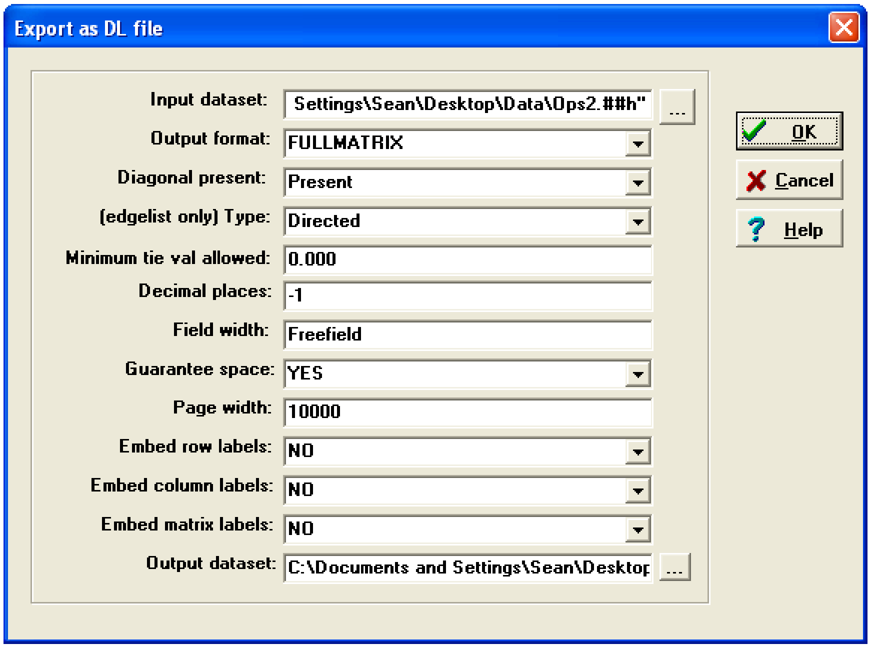


Figure 4.10: UCINET Export>DL dialog box

It is possible that you will get an error message the first time you click OK. It usually works the second time, but you will probably need to change the extension to .dat again. Note that when exporting two-mode data, UCINET does not provide a "Launch Pajek" option. Thus, you will need to open Pajek separately and read the network manually.

Warning: Exporting One and Two-mode Data to Pajek

[Pajek – Main Screen]

Net>Transform >Arcs→Edges>All

One thing to keep in mind when exporting data from UCINET to Pajek, regardless of whether you use UCINET's *Data>Export>Pajek>Network* command or its *Data>Export>DL* command, when Pajek reads the file into memory, it reads all ties as arcs (not edges). This is fine when you are working with directional data, but if you are working with nondirectional data (e.g., Noordin's Top's Friendship network), you will need to use Pajek's *Net> Transform>Arcs→Edges>All* command. This brings up a dialog box asking if you want to create a new network. This is generally a good idea so that you do not overwrite the original, so click OK. This brings up another dialog box asking if you want to remove multiple lines. Select option five (single line) and click OK. You should now have network that consists only of edges, not arcs.

Deriving One-Mode Social Networks from Two-Mode Social Networks

We can derive two one-mode networks (i.e., an actor-by-actor – “co-membership” – network and an event-by-event – “event overlap” – network) from a two-mode network by multiplying the original affiliation matrix by its transpose. Thankfully, both UCINET and Pajek have made this process relatively simple. Before turning to the method, however, let us (using Davis’s Southern Club Women as an example) first look at some of the interesting properties that such derived networks posses (Breiger 1974). Figure 4.11 (next page) displays the co-membership network. The diagonal tells us how many total events that each of the women attended. Evelyn attended eight events, Laura seven, Theresa eight, Brenda seven, and so on. Dorothy, Olivia and Flora attended the fewest events (two). The co-membership network also tells us how many events two women attended together. For example, Evelyn and Laura attended six events together, while Evelyn and Flora only attended one. Indeed, at a glance you can see that Evelyn attended at least one event with every other woman in the network, while Olivia and Flora didn’t attend any common event with Laura, Brenda, Charlotte, Frances and Eleanor.

		1 E	2 L	3 T	4 B	5 C	6 F	7 E	8 P	9 R	10 V	11 M	12 K	13 S	14 N	15 H	16 D	17 O	18 F
1	EVELYN	8	6	7	6	3	4	3	3	3	2	2	2	2	2	1	2	1	1
2	LAURA	6	7	6	6	3	4	4	2	3	2	1	1	2	2	2	1	0	0
3	THERESA	7	6	8	6	4	4	4	3	4	3	2	2	3	3	2	2	1	1
4	BRENDA	6	6	6	7	4	4	4	2	3	2	1	1	2	2	2	1	0	0
5	CHARLOTTE	3	3	4	4	4	2	2	0	2	1	0	0	1	1	1	0	0	0
6	FRANCES	4	4	4	4	2	4	3	2	2	1	1	1	1	1	1	1	0	0
7	ELEANOR	3	4	4	4	2	3	4	2	3	2	1	1	2	2	2	1	0	0
8	PEARL	3	2	3	2	0	2	2	3	2	2	2	2	2	2	1	2	1	1
9	RUTH	3	3	4	3	2	2	3	2	4	3	2	2	3	2	2	2	1	1
10	VERNE	2	2	3	2	1	1	2	2	3	4	3	3	4	3	3	2	1	1
11	MYRNA	2	1	2	1	0	1	1	2	2	3	4	4	4	3	3	2	1	1
12	KATHERINE	2	1	2	1	0	1	1	2	2	3	4	6	6	5	3	2	1	1
13	SYLVIA	2	2	3	2	1	1	2	2	3	4	4	6	7	6	4	2	1	1
14	NORA	2	2	3	2	1	1	2	2	2	3	3	5	6	8	4	1	2	2
15	HELEN	1	2	2	2	1	1	2	1	2	3	3	3	4	4	5	1	1	1
16	DOROTHY	2	1	2	1	0	1	1	2	2	2	2	2	2	1	1	2	1	1
17	OLIVIA	1	0	1	0	0	0	0	1	1	1	1	1	1	2	1	1	2	2
18	FLORA	1	0	1	0	0	0	0	1	1	1	1	1	1	2	1	1	2	2

Figure 4.11: Southern Club Women: Co-Membership Network

The event-overlap network (Figure 4.12) presents useful information as well. Here, the diagonal tells us how many women attended each even. In other words, three women attended events E1, E2, E13 and E14, while events E7 (ten women), E8 (fourteen women) and E9 (twelve women) were by far the most popular. The off-diagonal cells tell us how many women each event “shared.” Thus, events E1

and E2 shared two women (i.e., two women attended both E1 and E2), while events E8 and E9 shared 9.

		1 E1	2 E2	3 E3	4 E4	5 E5	6 E6	7 E7	8 E8	9 E9	10 E1	11 E1	12 E1	13 E1	14 E1
1	E1	3	2	3	2	3	3	2	3	1	0	0	0	0	0
2	E2	2	3	3	2	3	3	2	3	2	0	0	0	0	0
3	E3	3	3	6	4	6	5	4	5	2	0	0	0	0	0
4	E4	2	2	4	4	4	3	3	3	2	0	0	0	0	0
5	E5	3	3	6	4	8	6	6	7	3	0	0	0	0	0
6	E6	3	3	5	3	6	8	5	7	4	1	1	1	1	1
7	E7	2	2	4	3	6	5	10	8	5	3	2	4	2	2
8	E8	3	3	5	3	7	7	8	14	9	4	1	5	2	2
9	E9	1	2	2	2	3	4	5	9	12	4	3	5	3	3
10	E10	0	0	0	0	0	1	3	4	4	5	2	5	3	3
11	E11	0	0	0	0	0	1	2	1	3	2	4	2	1	1
12	E12	0	0	0	0	0	1	4	5	5	5	2	6	3	3
13	E13	0	0	0	0	0	1	2	2	3	3	1	3	3	3
14	E14	0	0	0	0	0	1	2	2	3	3	1	3	3	3

Figure 4.12: Southern Club Women: Event Overlap Network

In other words by simply transforming a two-mode network into a one-mode network, we create helpful information about the network we are examining, long before we estimate more complicated metrics.

UCINET

Data>Affiliations

In UCINET you derive one-mode networks from two-mode networks using the *Data>Affiliations* command. This brings up a dialog box like the one illustrated in Figure 4.13. For an actor-by actor matrix, choose “Row” in the “Which mode” option since actors are generally listed in rows; for an event-by-event matrix, choose “Column” in the “Which mode” option since events (affiliations) are generally listed in columns.

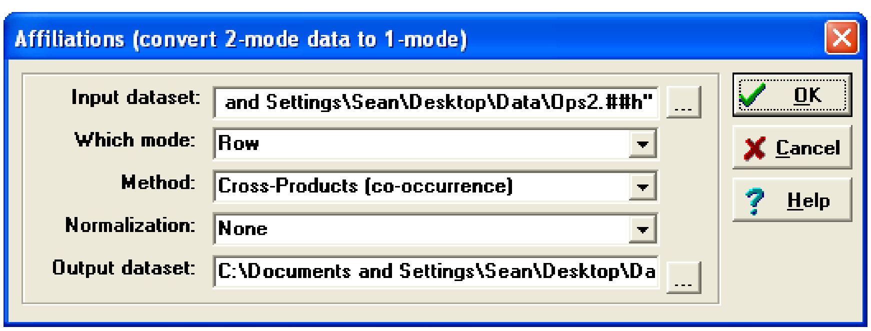


Figure 4.13: UCINET Affiliations dialog box

Be sure to save the files (“Output dataset”) under different file names, otherwise if you derive both an actor-by-actor matrix and an event-by-event matrix, whichever one you derive last will overwrite any ones you derived earlier.

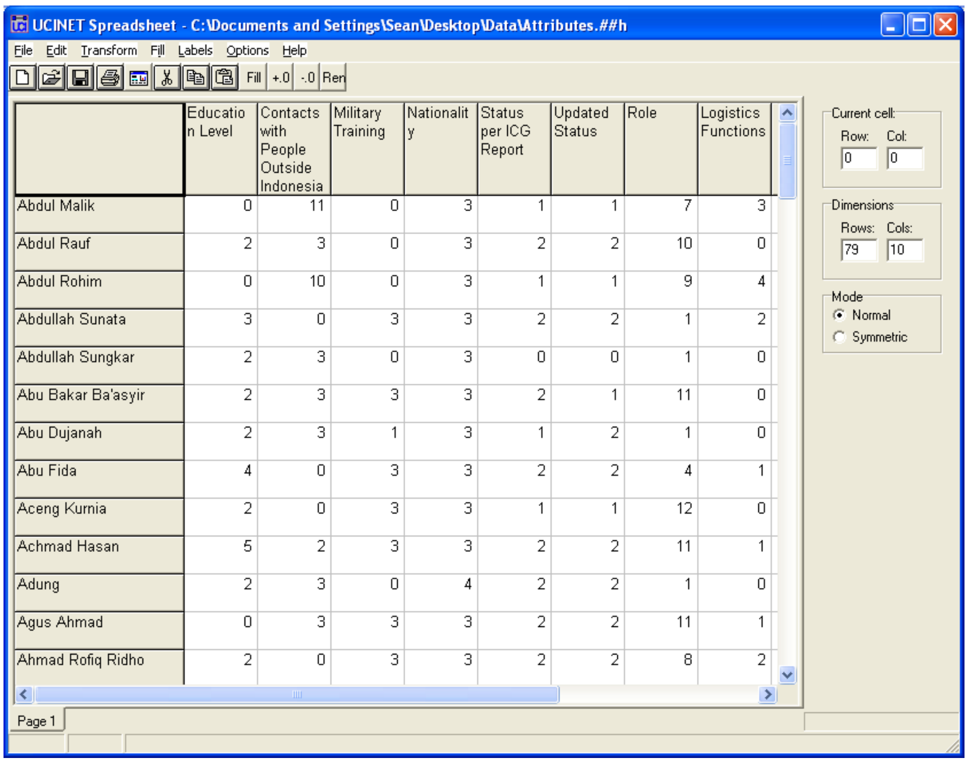
Data>Display

Unfortunately, a warning box does not appear in UCINET when you are about to overwrite an already existing file. You can display the two newly-created networks by either choosing the *Display* option found under the *Data* menu or by clicking on "D" icon located just below UCINET's menu bar.

Pajek

Net>Transform >2-Mode to 1-Mode >Rows, Columns

Deriving one-mode networks from two-mode networks in Pajek is simple, but (like in UCINET) you need to know which actors/events are assigned to the rows and columns. To create an actor-by-actor (co-membership) matrix choose the *Rows* option under the *Net>Transform>2-Mode to 1-Mode* submenu (assuming that that actors appear are listed in rows). After issuing the command, the Report window will appear. Close this and you will see that a new network appears in the Network drop list. Repeat the procedure, except choose the *Columns* option.



	Education Level	Contacts with People Outside Indonesia	Military Training	Nationality	Status per ICG Report	Updated Status	Role	Logistics Functions
Abdul Malik	0	11	0	3	1	1	7	3
Abdul Rauf	2	3	0	3	2	2	10	0
Abdul Rohim	0	10	0	3	1	1	9	4
Abdullah Sunata	3	0	3	3	2	2	1	2
Abdullah Sungkar	2	3	0	3	0	0	1	0
Abu Bakar Ba'asyir	2	3	3	3	2	1	11	0
Abu Dujanah	2	3	1	3	1	2	1	0
Abu Fida	4	0	3	3	2	2	4	1
Aceng Kurnia	2	0	3	3	1	1	12	0
Achmad Hasan	5	2	3	3	2	2	11	1
Adung	2	3	0	4	2	2	1	0
Agus Ahmad	0	3	3	3	2	2	11	1
Ahmad Rofiq Ridho	2	0	3	3	2	2	8	2

Figure 4.13: Noordin Attribute Data

Attribute Data

As we discussed in the second chapter, although social network analysis focuses primarily on the pattern of ties between actors, it does not ignore actors attribute data. Recording attribute data in UCINET is relatively straightforward.

Figure 4.14 (previous page) displays attribute data related to Noordin's Network. As you can see, the names of actors appear in rows while the various types of attributes appear in columns. These data differ from social network data in that each column contains self-contained. For example, the seventh column indicates the various roles that individuals within Noordin's network filled with each number indicating a different role. Obviously, without a code book (see Appendix 1), there is no way to know what each of the various numbers mean. We can use attribute data in various ways, some of which we will explore later in the guide. For now, it is only necessary to demonstrate how it is recorded.

4.6 Aggregating Networks (Matrices)

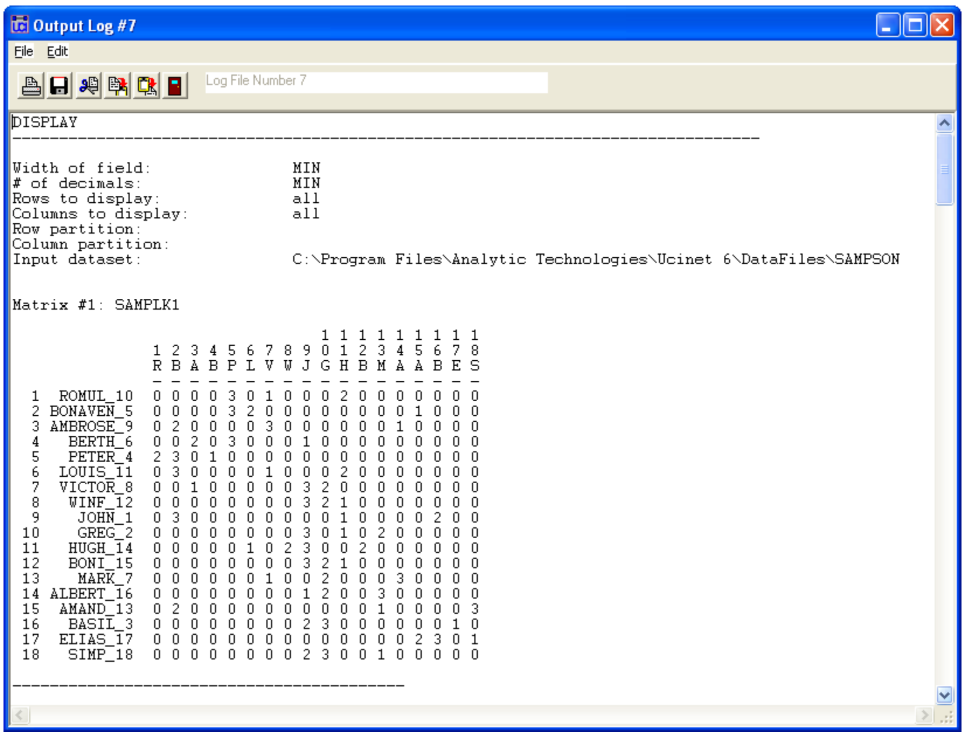
Up to this point we have focused on single types of relationship among actors: friendship, kinship, operations, etc. In the real world, however, actors are typically involved in more than one type of relation (Hanneman and Riddle 2005). As we have already seen (Chapter 1), most individuals are embedded in a number of types of ties (e.g., friendship, kinship and economic), and corporate and state actors are no different. Businesses engage in financial and informational exchanges and sometimes form alliances with one another (Saxenian 1994), while countries are linked through numerous cultural, economic, military and political ties not to mention transnational corporations, nongovernmental organizations and international agencies (Meyer et al. 1997). More importantly for our purposes, different ties often push and pull actors in different directions (Simmel [1908, 1922] 1955). Thus, coding multiple relations is especially important. UCINET includes a variety of tools for analyzing multiplex data. Some allow you to remove individual matrices from a multiple matrix file, while others allow you to create stacked social network data from separate single social network data files.

Multiple Relations in UCINET and NetDraw

[UCINET] Data>Display

One of the most common ways of storing multiplex data is by "stacking" a set of actor-by-actor matrices, one for each type of relation. Figure 4.13 (next page) displays part of the output from a *Data>Display* command for the Sampson Monastery data set (SAMPSON.##h). Sampson recorded the social interactions among a group of monks and collected numerous sociometric rankings. During his stay a "crisis in the cloister" developed that resulted in the expulsion of four monks and the voluntary departure of several others. In the end, only four remained in the monastery (Bonaventure, Berthold, Ambrose and Louis).

[UCINET] Sampson coded four types of relations, with separate matrices indicating positive and negative ties. Each monk ranked only his top three choices on that tie. The relations are esteem (SAMPES) and disesteem (SAMPDES), liking (SAMPLK – three different time periods were recorded) and disliking (SAMPDLK – only one time period), positive influence (SAMPIN) and negative influence (SAMPNIN), praise (SAMPPR) and blame (SAMPNPR). In each matrix 3 indicates the highest or first choice and 1 the last choice. (Some subjects offered tied ranks for their top four choices). *Data>Display* Display the file yourself in order to see the complete output.[19] *Data>Spreadsheets >Matrix>File>Open* Next, using UCINET's spreadsheet function, open the file as a spreadsheet and note that each matrix is stored on a separate sheet.



```
DISPLAY
--------------------------------------------------------------------------------

Width of field:               MIN
# of decimals:                MIN
Rows to display:              all
Columns to display:           all
Row partition:
Column partition:
Input dataset:                C:\Program Files\Analytic Technologies\Ucinet 6\DataFiles\SAMPSON


Matrix #1: SAMPLK1

                                    1 1 1 1 1 1 1 1 1
               1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8
               R B A B P L V W J G H B M A A B E S
               - - - - - - - - - - - - - - - - - -
  1  ROMUL_10  0 0 0 0 3 0 1 0 0 0 2 0 0 0 0 0 0 0
  2 BONAVEN_5  0 0 0 0 3 2 0 0 0 0 0 0 0 0 1 0 0 0
  3 AMBROSE_9  0 2 0 0 0 0 3 0 0 0 0 0 0 1 0 0 0 0
  4   BERTH_6  0 0 2 0 3 0 0 0 1 0 0 0 0 0 0 0 0 0
  5   PETER_4  2 3 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
  6  LOUIS_11  0 3 0 0 0 0 1 0 0 0 2 0 0 0 0 0 0 0
  7  VICTOR_8  0 0 1 0 0 0 0 0 3 2 0 0 0 0 0 0 0 0
  8   WINF_12  0 0 0 0 0 0 0 0 3 2 1 0 0 0 0 0 0 0
  9    JOHN_1  0 3 0 0 0 0 0 0 0 0 1 0 0 0 0 2 0 0
 10    GREG_2  0 0 0 0 0 0 0 0 3 0 1 0 2 0 0 0 0 0
 11   HUGH_14  0 0 0 0 0 1 0 2 3 0 0 2 0 0 0 0 0 0
 12   BONI_15  0 0 0 0 0 0 0 0 3 2 1 0 0 0 0 0 0 0
 13    MARK_7  0 0 0 0 0 0 1 0 0 2 0 0 0 3 0 0 0 0
 14 ALBERT_16  0 0 0 0 0 0 0 0 1 2 0 0 3 0 0 0 0 0
 15  AMAND_13  0 2 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 3
 16   BASIL_3  0 0 0 0 0 0 0 0 2 3 0 0 0 0 0 0 1 0
 17  ELIAS_17  0 0 0 0 0 0 0 0 0 0 0 0 0 0 2 3 0 1
 18   SIMP_18  0 0 0 0 0 0 0 0 2 3 0 0 1 0 0 0 0 0

-------------------------------------------
```

Figure 4.14: UCINET Output Log displaying Sampson's Monastery Data

NetDraw

Before looking at the variety of tools UCINET has for analyzing multiplex data, let's first visualize the Sampson data using NetDraw. As we saw in Chapter 3, NetDraw allows users to use multiple lines (with different colors or styles) and overlay one relation on top of another if the data are stacked as multiple matrices within the same file (e.g., like the Sampson data). Open the "SAMPSON.##h"

[19] The Sampson data is one of the standard datasets that comes with UCINET, so you can find it in the UCINET data files folder. Also, under UCINET's Help function (sample datasets), you can find a more complete description of the Sampson data.

[NetDraw] File>Open >Ucinet dataset>Network

dataset, using the *File>Open>Ucinet dataset>Network* command. Recall that the "Rels" tab (see Figure 4.15 below) allows you to select which network to view, which can be useful for combining and switching back and forth between relations.

Properties>Lines>Color >Relation

You can also assign different colors to the various relations by using the dialog box (not shown) that the *Properties>Lines>Color>Relation* command calls up. There are other ways to color relations in NetDraw, but this seems to be the most stable approach (at least for now).

UCINET

[UCINET] Data>Unpack

Next, let us examine at how to unpack stacked network data in UCINET. To do this in UCINET simply choose the *Data>Unpack* command. This brings up an "Unpack" dialog box (Figure 4.16) that asks you for the input dataset and which relations to unpack. You can choose to unpack all of the relations or just some. UCINET's default is "All" but if you wanted to only unpack some, you would click on the "L" radio button, which brings up an additional dialog box that lets you pick the matrices of your choice. In this case, choose "All" and UCINET will unpack ten separate matrices/networks.

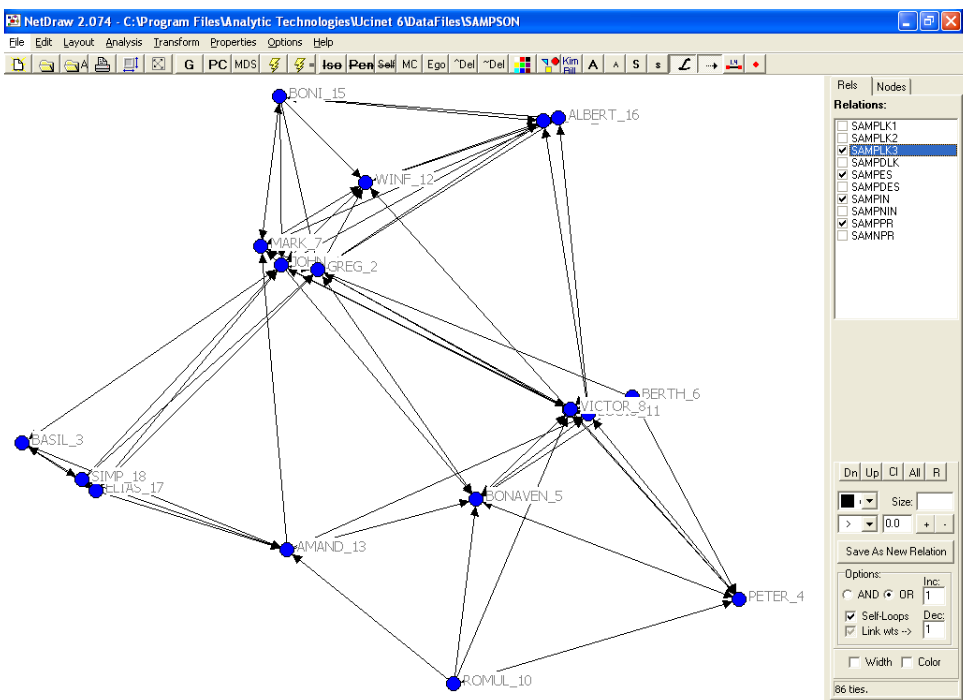


Figure 4.15: NetDraw drawing of Sampson's Monastery Data

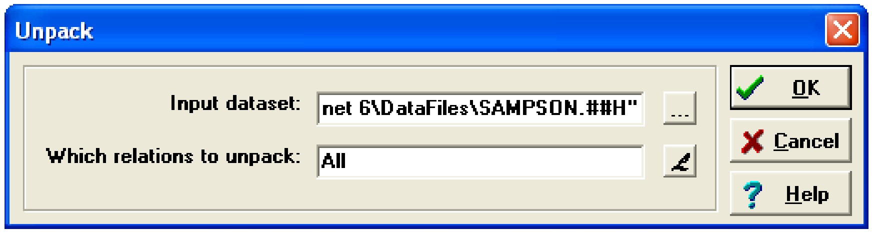


Figure 4.16: UCINET Unpack dialog box

It is easy in UCINET to combine separate matrices into a stacked dataset (i.e., just reverse what we did above), using the *Data>Join* command. This brings up a "Join" dialog box (Figure 4.17, next page). Here, I've chosen to rejoin esteem (SAMPES), liking (SAMPLK3), positive influence (SAMPIN) and praise (SAMPPR) into a single file called "Joined."

Data>Join

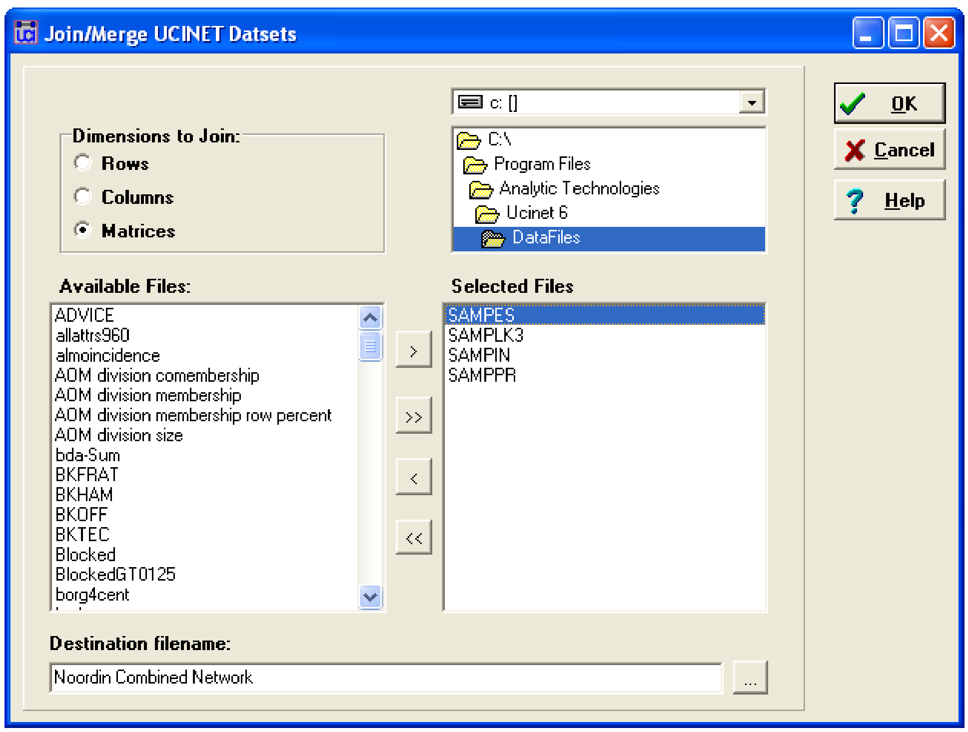


Figure 4.17: UCINET Join/Merge Dataset Dialog Box

Note that in the above example we chose "Matrices" under the "Dimensions to join" options on the upper left hand side of the dialog box. As this figure suggests, you can also combine the rows of two or more matrices (keeping the columns the same) or the columns of two or more matrices (keeping the rows the same). This command can be quite useful for combining and analyzing two-mode (i.e., affiliation) matrices.

Transform >Matrix Operations >Within dataset >Aggregations

You may want to create a single-valued matrix from a series of stacked matrices (like the Sampson data we just joined). To do this we need to use the *Transform>Matrix Operations>Within dataset>Aggregations* command, which brings up the following dialog box (Figure 4.18).

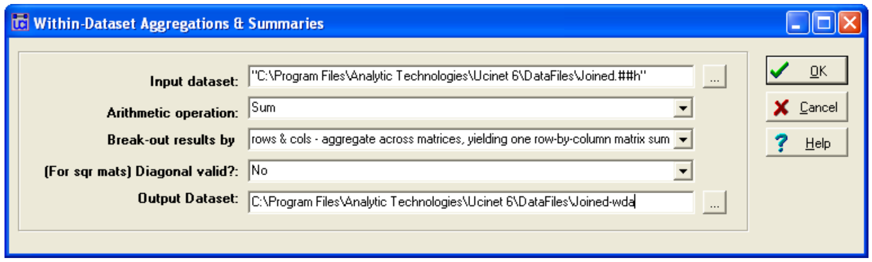


Figure 4.18: UCINET Within Dataset Aggregations dialog box

Data>Display

After issuing this command, use the *Data>Display* command to see whether this worked as expected.

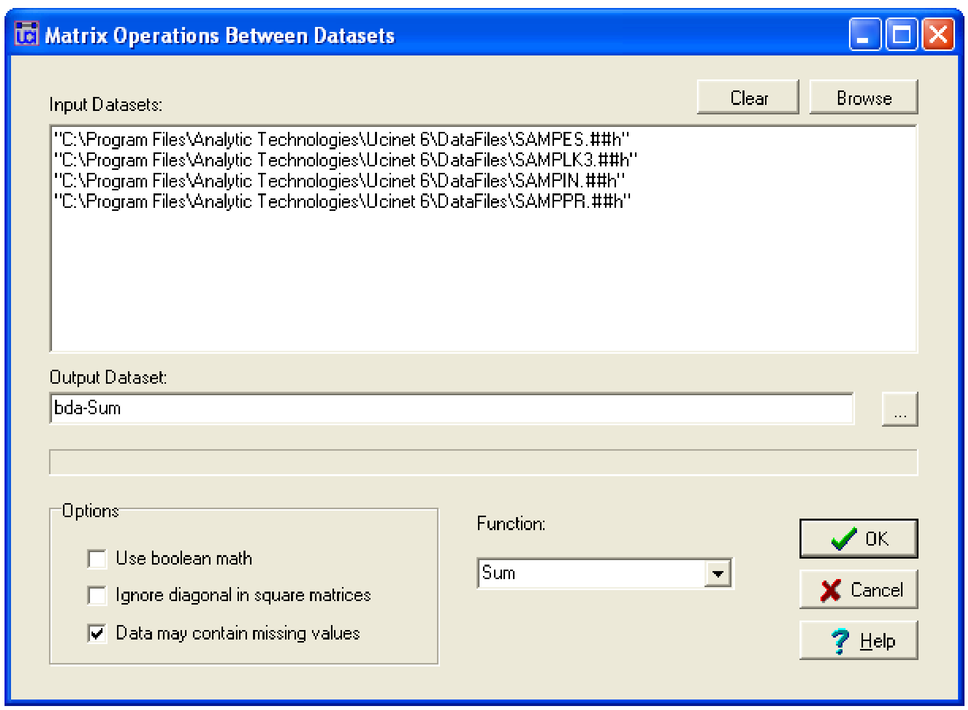


Figure 4.19: UCINET Between Dataset Statistical Summaries Dialog Box

Transform >Matrix Operations >Between datasets >Statistical Summaries

Next, let us assume that we have not yet combined various matrices into a single, stacked dataset, but you still want to combine (i.e., sum) two or more matrices into a single-valued matrix. UCINET allows you to do this as well (although it does not always work unfortunately) using the *Transform>Matrix Operations>Between datasets>Statistical Summaries* command, which brings up a dialog box (Figure 4.19, previous page). Click OK and UCINET combines the matrices into a single-matrix and displays a single-valued matrix in an output log.

Aggregating Noordin's Networks

It is time to apply some of these techniques (and some variations on them) to Noordin's Networks. We will begin by joining the following types of ties found within Noordin's network:

- Business & Financial
- Friendship
- Internal Communication
- Kinship
- Logistical Place
- Operational
- Organizational
- Religious
- School
- Training

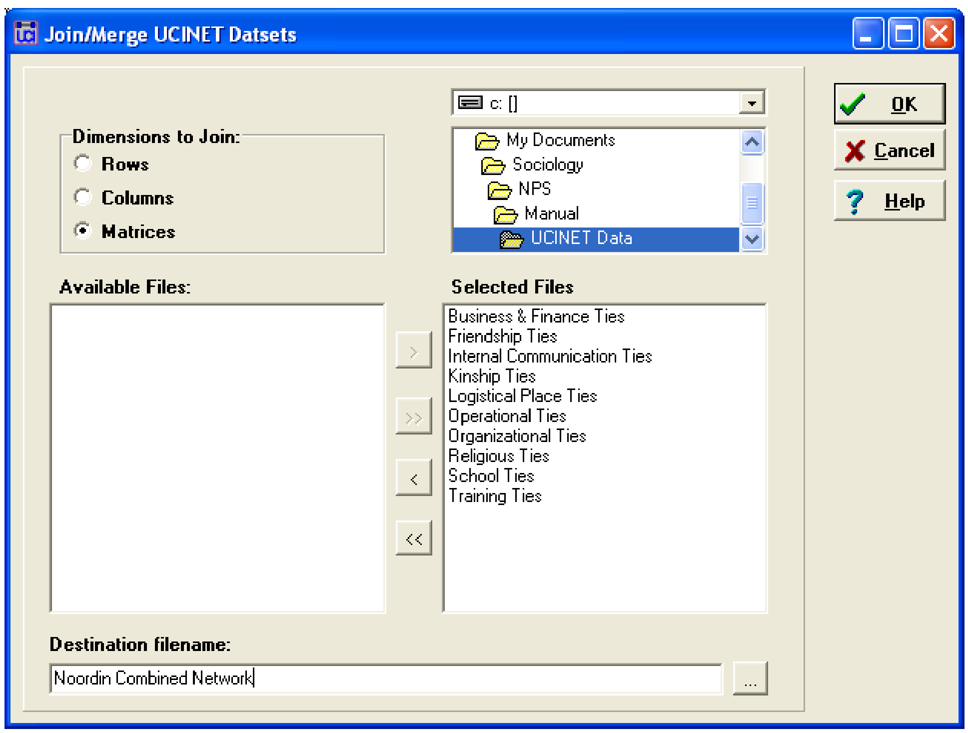


Figure 4.20: UCINET's Join/Merge Datasets Dialog Box

The friendship, internal communication and kinship networks were originally coded as one-mode networks, while the other seven were derived from networks originally coded as two-mode networks (see Appendix 1 for more complete discussion of these networks). We have already seen how easy it is in UCINET to

combine separate matrices into a stacked dataset using its *Data>Join* command. Issue this command again, which brings up UCINET's Join/Merge Dialog Box (Figure 4.20, previous page). After selecting OK, an output log appears (not shown) that displays each of the stacked matrices (you will need to scroll down the output log). You can also examine the separate matrices using UCINET's internal spreadsheet program (Figure 4.21), which is accessed using either the spreadsheet icon on the menu bar or the *Data>Spreadsheets>Matrix>File>Open* command. The tabs running along the bottom of the spreadsheet indicate that each network has its own sheet, which can be accessed (and edited) just as you would in a standard spreadsheet program such as Excel. Figure 4.22 (next page) displays the aggregated network in NetDraw.

Data>Spreadsheets >Matrix>File>Open

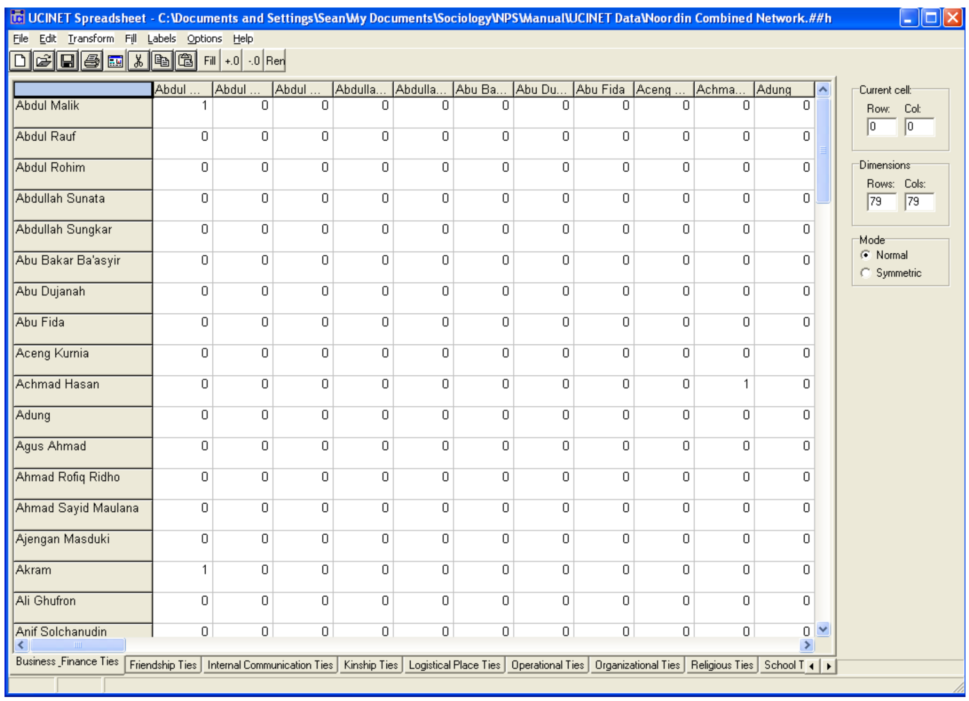


Figure 4.21: UCINET's Spreadsheet Program

Transform >Matrix Operations >Within dataset >Aggregations

To create a single-valued matrix from these stacked matrices we use the *Transform>Matrix Operations>Within dataset>Aggregations* command, which brings up a dialog box (Figure 4.18, above). Selecting OK produces an output log similar to the one in Figure 4.23 below. Note that some of the cells in the matrix have values greater than one. This indicates how many different types of ties each pair of actors has with one another. For example, Abdullah Sungkar and Abu Bakar Ba'asyir share four different types of ties. A quick examination of the ten individual networks indicates that these are made up of friendship, school and two organizational ties.

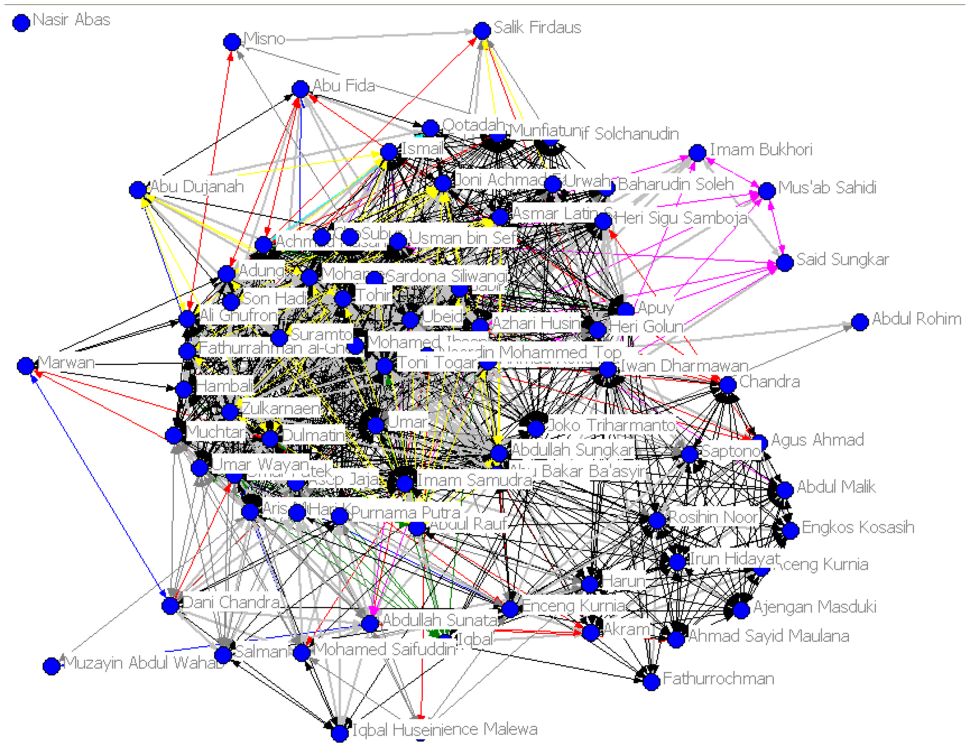
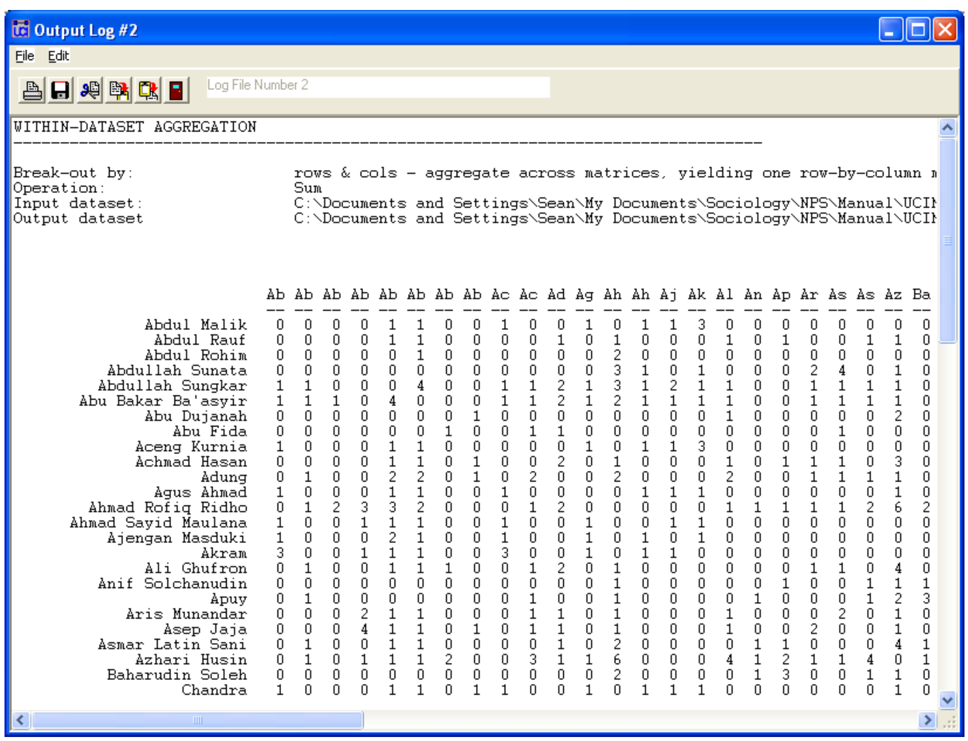


Figure 4.22: Network Map of Noordin's Aggregated Network



WITHIN-DATASET AGGREGATION

Break-out by:	rows & cols - aggregate across matrices, yielding one row-by-column m
Operation:	Sum
Input dataset:	C:\Documents and Settings\Sean\My Documents\Sociology\NPS\Manual\UCIN
Output dataset	C:\Documents and Settings\Sean\My Documents\Sociology\NPS\Manual\UCIN

	Ab	Ab	Ab	Ab	Ab	Ab	Ab	Ab	Ac	Ac	Ad	Ag	Ah	Ah	Aj	Ak	Al	An	Ap	Ar	As	As	Az	Ba
Abdul Malik	0	0	0	0	1	1	0	0	1	0	0	1	0	1	1	3	0	0	0	0	0	0	0	0
Abdul Rauf	0	0	0	0	1	1	0	0	0	0	1	0	1	0	0	0	1	0	1	0	0	1	1	0
Abdul Rohim	0	0	0	0	0	1	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0
Abdullah Sunata	0	0	0	0	0	0	0	0	0	0	0	0	3	1	0	1	0	0	0	2	4	0	1	0
Abdullah Sungkar	1	1	0	0	0	4	0	0	1	1	2	1	3	1	2	1	1	0	0	1	1	1	1	0
Abu Bakar Ba'asyir	1	1	1	0	4	0	0	0	1	1	2	1	2	1	1	1	1	0	0	1	1	1	1	0
Abu Dujanah	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0	0	0	2	0
Abu Fida	0	0	0	0	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0	0	1	0	0	0
Aceng Kurnia	1	0	0	0	1	1	0	0	0	0	0	1	0	1	1	3	0	0	0	0	0	0	0	0
Achmad Hasan	0	0	0	0	1	1	0	1	0	0	2	0	1	0	0	0	1	0	1	1	1	0	3	0
Adung	0	1	0	0	2	2	0	1	0	2	0	0	2	0	0	0	2	0	0	1	1	1	1	0
Agus Ahmad	1	0	0	0	1	1	0	0	1	0	0	0	0	1	1	1	0	0	0	0	0	0	1	0
Ahmad Rofiq Ridho	0	1	2	3	3	2	0	0	0	1	2	0	0	0	0	0	1	1	1	1	1	2	6	2
Ahmad Sayid Maulana	1	0	0	1	1	1	0	0	1	0	0	1	0	0	1	1	0	0	0	0	0	0	0	0
Ajengan Masduki	1	0	0	0	2	1	0	0	1	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0
Akram	3	0	0	1	1	1	0	0	3	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0
Ali Ghufron	0	1	0	0	1	1	1	0	0	1	2	0	1	0	0	0	0	0	0	1	1	0	4	0
Anif Solchanudin	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	1	1	1
Apuy	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	1	2	3
Aris Munandar	0	0	0	2	1	1	0	0	0	1	1	0	1	0	0	0	1	0	0	0	2	0	1	0
Asep Jaja	0	0	0	4	1	1	0	1	0	1	1	0	1	0	0	0	1	0	0	2	0	0	1	0
Asmar Latin Sani	0	1	0	0	1	1	0	0	0	0	1	0	2	0	0	0	0	1	1	0	0	0	4	1
Azhari Husin	0	1	0	1	1	1	2	0	0	3	1	1	6	0	0	0	4	1	2	1	1	4	0	1
Baharudin Soleh	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	1	3	0	0	1	1	0
Chandra	1	0	0	0	1	1	0	1	1	0	0	1	0	1	1	1	0	0	0	0	0	0	1	0

Figure 4.23: UCINET Output Log of Aggregated Matrix

Transform
>Matrix Operations
>Between datasets
>Statistical Summaries

You can also use the *Data>Display* command to examine the aggregated matrix. If all we wanted was an aggregated matrix, we did not have to first stack the ten matrices. Instead, we could have aggregated the data using UCINET's *Transform>Matrix Operations>Between datasets>Statistical Summaries* command (not shown – see Figure 4.19 above). The advantage of stacking the matrices is that it allows us to use some of NetDraw's more helpful features (e.g., assigning different colors to lines indicating different types of ties – see Figure 4.22 above).

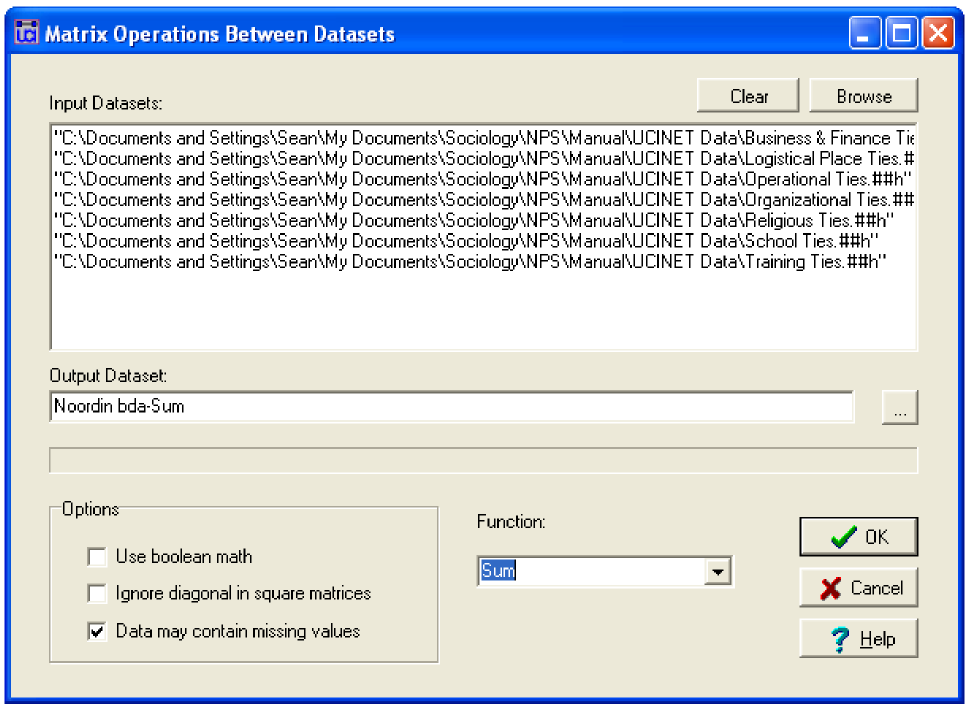


Figure 4.24: UCINET Between Dataset Statistical Summaries Dialog Box

Aggregating and Manipulating Noordin's Networks: A More Complex Example

Of course, just because two people are members of the same organization or attend the same event does not mean that they necessarily are friends or even know one another. Thus, we may want to use some sort of "membership" threshold before concluding that a tie actually exists between two actors. Looking at the Noordin data, let us assume for illustrative purposes that a tie exists between two actors if they are friends, kin or communicate with one another or they are members of or participated in two or more of the same groups or events. In other words, if two people only attended the same school, then we will not assume that a tie exists between them. However, if they attended the same school and are members of the same mosque (or participated in the same operation, are members

of the same organization, etc.) then we will assume that a tie does exist between them. How can we capture this assumption using social network analysis? We will first only aggregate Noordin's affiliation networks using UCINET's *Transform>Matrix Operations>Between datasets>Statistical Summaries* command (Figure 4.24, previous page).

Transform>Dichotomize

Next, we use UCINET's *Transform>Dichotomize* command (Figure 4.25) to assign a "1" for every pair of actors that shares two or more memberships and a "O" for every pair of actors that shares one or fewer memberships. This creates a new matrix of "0's" and "1's" (Figure 4.26, below).

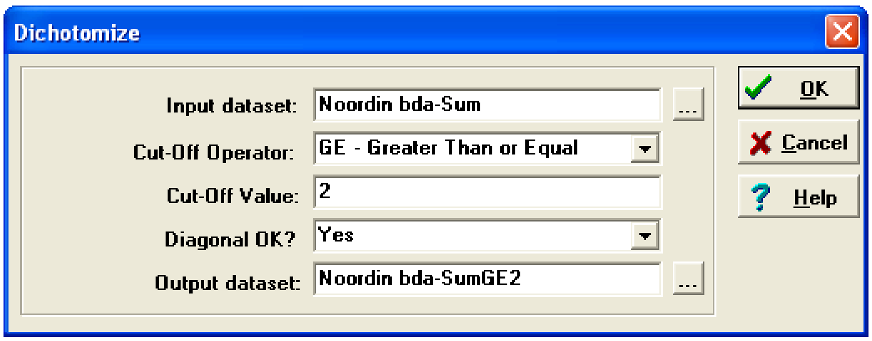


Figure 4.25: UCINET Dichotomize (Binarize) Dialog Box

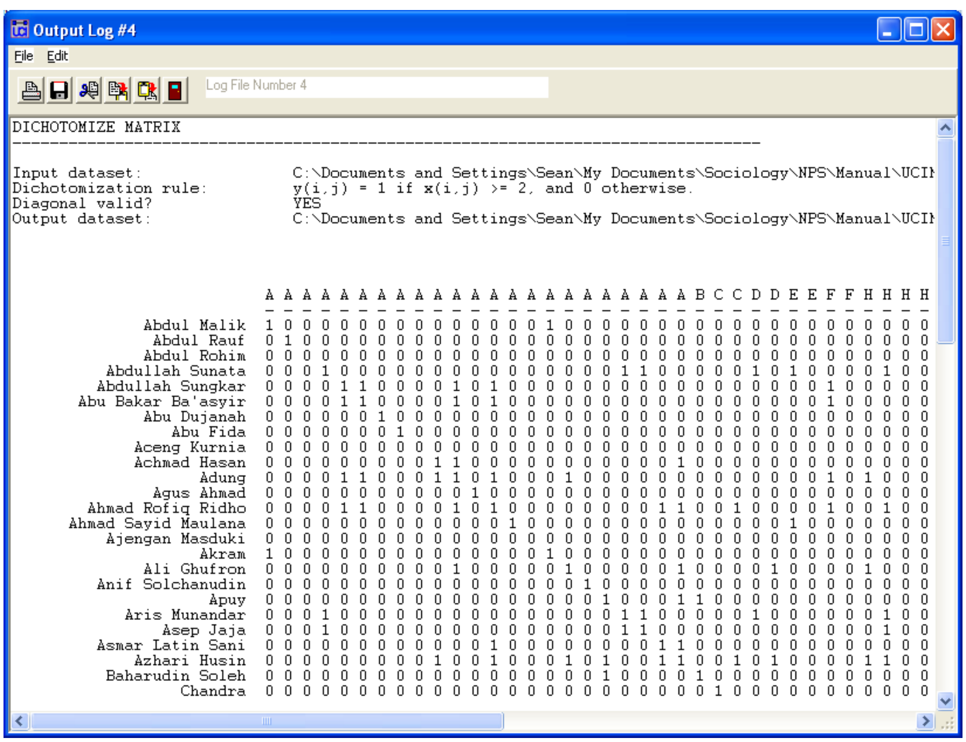


Figure 4.26: UCINET Output Log of Dichotomized Matrix

We can now stack with the friendship, kinship and internal communication networks by using UCINET's *Data>Join* command (not shown), which results in the network map (created by NetDraw) shown in Figure 4.27. As you can see this network is not nearly as dense as the one shown in Figure 4.22, which makes sense since we effectively treated all affiliation ties between actors as only one tie, and that was only the case if they shared two or more affiliations. Obviously, we could have used different assumptions in creating this modified map of Noordin's network, but that is beside the point here. All we are interested in showing at this juncture is how we can manipulate various networks in different ways to reflect different assumptions.

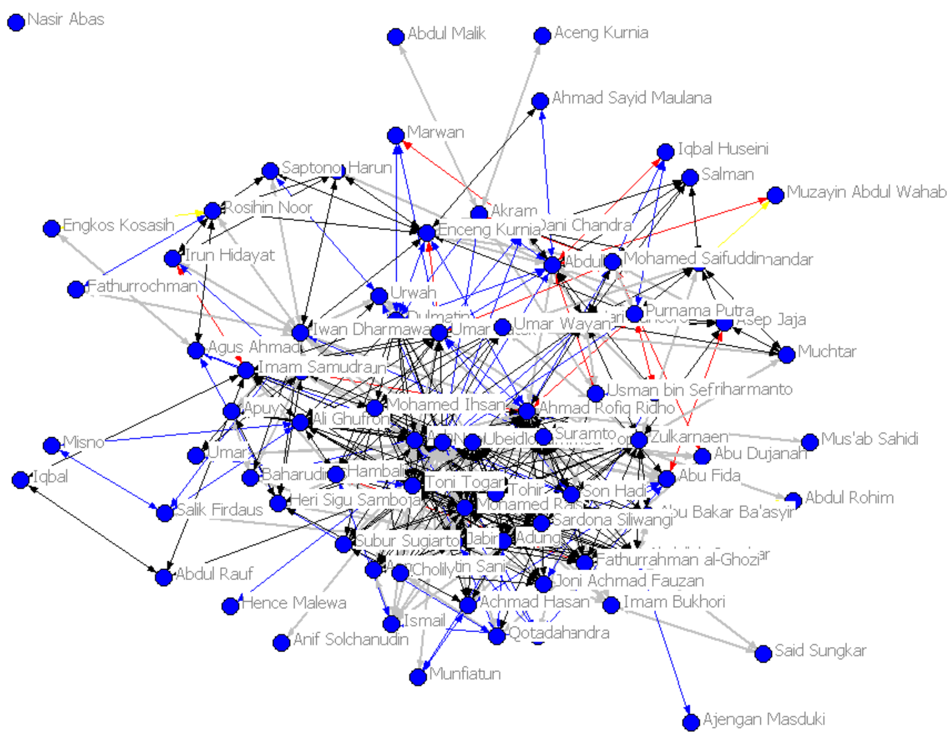


Figure 4.27: Network Map of Noordin's Aggregated Network

Multiple Relations in Pajek

With version 1.02 Pajek added a feature that allows users to work with multiplex data. How does Pajek store multirelational network data? It does so by allowing us to assign relation numbers to a set of ties (i.e., arcs, edges). Figure 4.28 (Padgett marriage and business networks) illustrates how such data are recorded and stored. As you can see that relation numbers and names are added to the edges (or arcs) statements after the list of vertices (i.e., nodes).

Padgett.net - Notepad

File Edit Format View Help

```
*Vertices 16
  1 "ACCIAIUOL"                                  0.1570    0.9173    0.5000
  2 "ALBIZZI"                                    0.6720    0.4732    0.5000
  3 "BARBADORI"                                  0.4132    0.5244    0.5000
  4 "BISCHERI"                                   0.3489    0.1427    0.5000
  5 "CASTELLAN"                                  0.3403    0.3859    0.5000
  6 "GINORI"                                     0.6692    0.6832    0.5000
  7 "GUADAGNI"                                   0.4748    0.2370    0.5000
  8 "LAMBERTES"                                  0.5642    0.1500    0.5000
  9 "MEDICI"                                     0.3813    0.6620    0.5000
 10 "PAZZI"                                      0.5147    0.9590    0.5000
 11 "PERUZZI"                                    0.2853    0.2535    0.5000
 12 "PUCCI"                                      0.9133    0.8974    0.5000
 13 "RIDOLFI"                                    0.0410    0.5748    0.5000
 14 "SALVIATI"                                   0.3895    0.7924    0.5000
 15 "STROZZI"                                    0.0582    0.2573    0.5000
 16 "TORNABUON"                                  0.2740    0.5842    0.5000
*Edges :1 "Marriage"
  1   9 1
  2   6 1
  2   7 1
  2   9 1
  3   5 1
  3   9 1
  4   7 1
  4  11 1
  4  15 1
  5  11 1
  5  15 1
  7   8 1
  7  16 1
  9  13 1
  9  14 1
  9  16 1
 10  14 1
 11  15 1
 13  15 1
 13  16 1
*Edges :2 "Business"
  3   5 1
  3   6 1
  3   9 1
```

Figure 4.28: Padgett Marriage and Business Network Data in Pajek Format

[Pajek - Draw Screen] Options>Colors>Edges >Relation Number

Options>Colors>Arcs >Relation Number

Options>Colors >Relation Colors

Options>Lines>Draw Lines>Relations

As with NetDraw we can represent the relation number of a line by line color, which can be done in the Draw screen with the *Options>Colors> Edges>Relation Number* (for edges) and *Options>Colors>Arcs>Relation Number* (for arcs) commands. We can choose the color of each relation number in the *Options> Colors>Relation Colors* dialog screen. We can also draw the lines of just one relation by selecting a particular relation number in the *Options>Lines> Draw Lines>Relations* dialog box. Figure 4.29 (next page) presents a network map of the Padgett data where marriage ties are colored red and the business ties are colored blue. When two actors share a tie (e.g., Bischeri and Peruzzi), Pajek colors the ties by the last relation in the file (in this case, business = blue). Recall that NetDraw colors multiple relations grey.

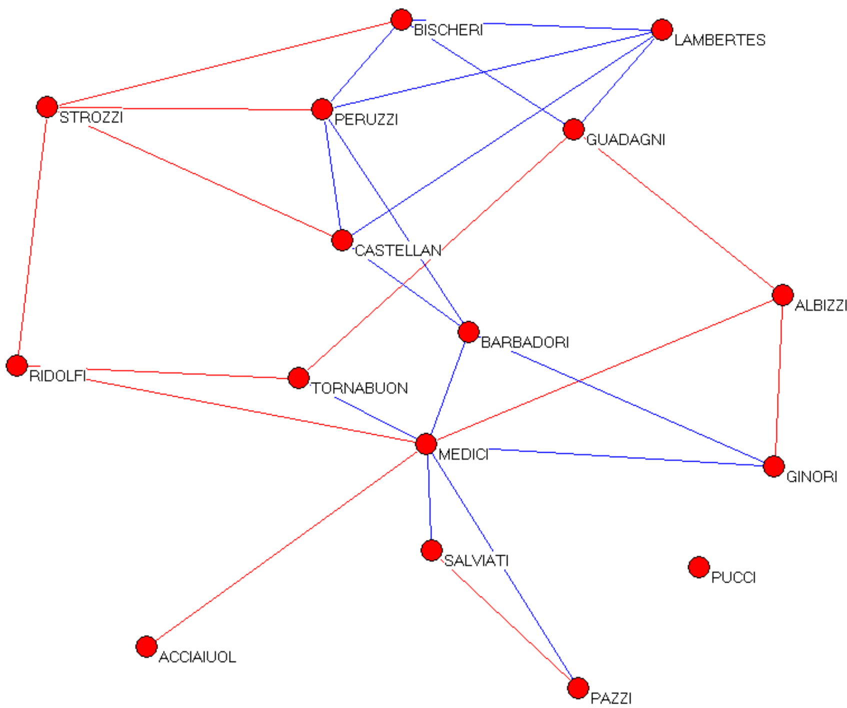


Figure 4.29: Network Map of Padgett Multirelational Data

[Main Screen]
Net>Transform
>Multiple Relations
>Extract Relations

Net>Transform
>Multiple Relations
>Change Relation Number

Pajek allows us to extract one or more relations from a multiple relations network with the *Net>Transform>Multiple Relations>Extract Relation(s)* command. When we issue this command Pajek generates a new network for each of the selected relation numbers, preserving the relation number. We can also recode relation number and change relation labels with the *Net>Transform> Multiple Relations>Change Relation Number-Label* command. We can also change the label name of a relation by opening and editing the Pajek file in a text editor such as Notepad. That is how the labels were added to Figure 4.28 above.

Noordin's Networks in Pajek

[UCINET]
Data>Export>DL

[Pajek]
File>Network>Save

To get at sense of how to work with such data in Pajek, we will export the Noordin Combined Network we aggregated in UCINET to Pajek using UCINET's *Data>Export>DL* command; this brings up the following dialog box (Figure 4.30, next page), which by now should be familiar to you. As before, you will want to change the output dataset extension from ".txt" to ".dat." After reading the ".dat" file in Noordin, save it as a ".net" file, using Pajek's *File>Network>Save* command.

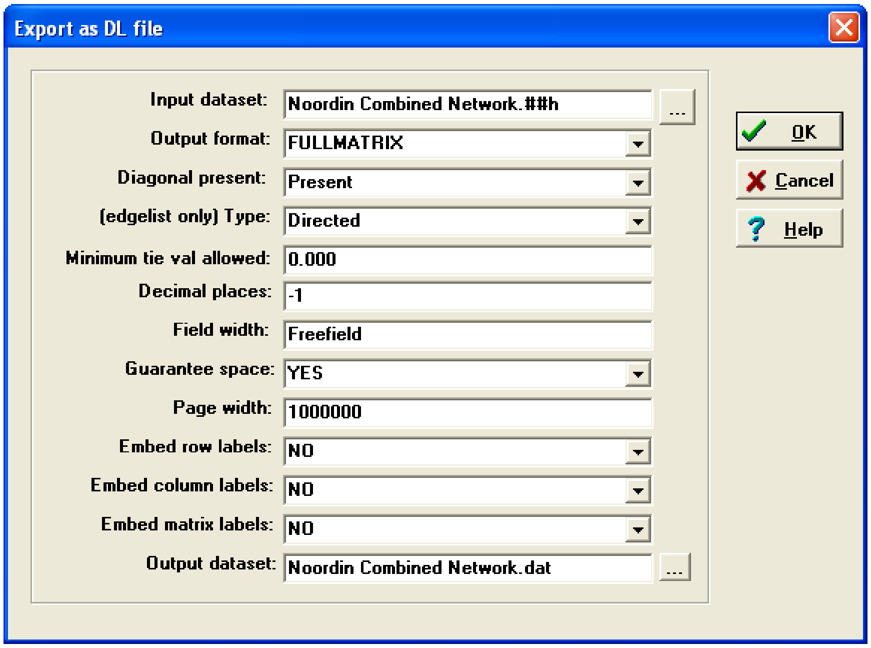
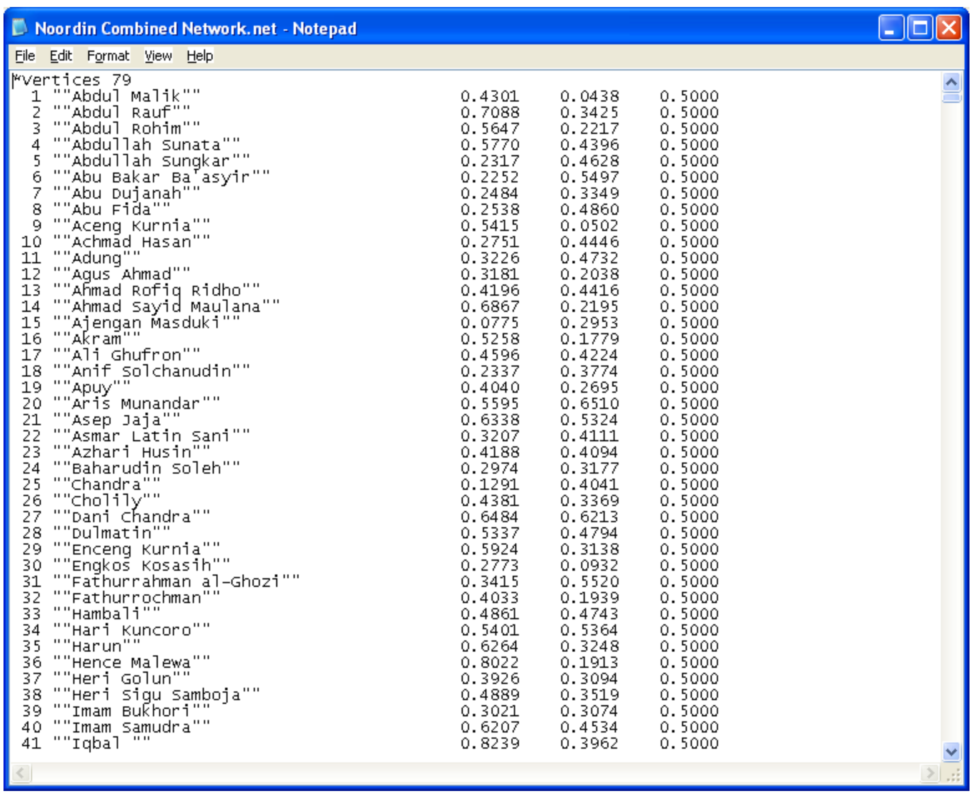
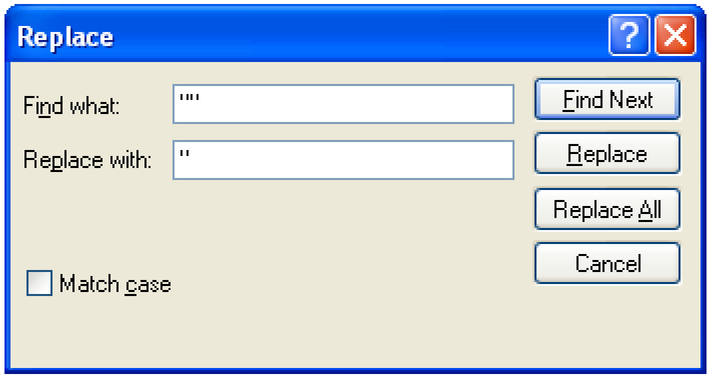


Figure 4.30: Network Map of Noordin's Aggregated Network




```
*Vertices 79
  1 ""Abdul Malik""                      0.4301    0.0438    0.5000
  2 ""Abdul Rauf""                       0.7088    0.3425    0.5000
  3 ""Abdul Rohim""                      0.5647    0.2217    0.5000
  4 ""Abdullah Sunata""                  0.5770    0.4396    0.5000
  5 ""Abdullah Sungkar""                 0.2317    0.4628    0.5000
  6 ""Abu Bakar Ba'asyir""               0.2252    0.5497    0.5000
  7 ""Abu Dujanah""                      0.2484    0.3349    0.5000
  8 ""Abu Fida""                         0.2538    0.4860    0.5000
  9 ""Aceng Kurnia""                     0.5415    0.0502    0.5000
 10 ""Achmad Hasan""                     0.2751    0.4446    0.5000
 11 ""Adung""                            0.3226    0.4732    0.5000
 12 ""Agus Ahmad""                       0.3181    0.2038    0.5000
 13 ""Ahmad Rofiq Ridho""                0.4196    0.4416    0.5000
 14 ""Ahmad Sayid Maulana""              0.6867    0.2195    0.5000
 15 ""Ajengan Masduki""                  0.0775    0.2953    0.5000
 16 ""Akram""                            0.5258    0.1779    0.5000
 17 ""Ali Ghufron""                      0.4596    0.4224    0.5000
 18 ""Anif Solchanudin""                 0.2337    0.3774    0.5000
 19 ""Apuy""                             0.4040    0.2695    0.5000
 20 ""Aris Munandar""                    0.5595    0.6510    0.5000
 21 ""Asep Jaja""                        0.6338    0.5324    0.5000
 22 ""Asmar Latin Sani""                 0.3207    0.4111    0.5000
 23 ""Azhari Husin""                     0.4188    0.4094    0.5000
 24 ""Baharudin Soleh""                  0.2974    0.3177    0.5000
 25 ""Chandra""                          0.1291    0.4041    0.5000
 26 ""Cholily""                          0.4381    0.3369    0.5000
 27 ""Dani Chandra""                     0.6484    0.6213    0.5000
 28 ""Dulmatin""                         0.5337    0.4794    0.5000
 29 ""Enceng Kurnia""                    0.5924    0.3138    0.5000
 30 ""Engkos Kosasih""                   0.2773    0.0932    0.5000
 31 ""Fathurrahman al-Ghozi""            0.3415    0.5520    0.5000
 32 ""Fathurrochman""                    0.4033    0.1939    0.5000
 33 ""Hambali""                          0.4861    0.4743    0.5000
 34 ""Hari Kuncoro""                     0.5401    0.5364    0.5000
 35 ""Harun""                            0.6264    0.3248    0.5000
 36 ""Hence Malewa""                     0.8022    0.1913    0.5000
 37 ""Heri Golun""                       0.3926    0.3094    0.5000
 38 ""Heri Sigu Samboja""                0.4889    0.3519    0.5000
 39 ""Imam Bukhori""                     0.3021    0.3074    0.5000
 40 ""Imam Samudra""                     0.6207    0.4534    0.5000
 41 ""Iqbal ""                           0.8239    0.3962    0.5000
```

Figure 4.31: Pajek Network File of Combined Noordin Network



[Notepad] Edit>Replace

Figure 4.32: Notepad Replace Dialog Box

If you close and reopen Pajek and then read in the "Noordin Combined Network.net" file, chances are the labels will be missing when you draw the network. This is because in the process of exporting and importing the network as a DL file and then saving it as a Pajek network file, Pajek inserts double-quotes (rather than single-quotes) around the vertex labels in the .net file (Figure 4.30, previous page). The easiest way to correct this is to open the file in Notepad and then use Notepad's "Find and Replace" function, telling Notepad to replace double quotes with single quotes (see Figure 4.32 above). *Before doing this, however*, you will want to also insert relation labels into the .net file (see Figure 4.28 above as an example). When you read the edited file back into Pajek, the vertex labels should be visible (assuming you saved your changes) (Figure 4.33). After telling Pajek to assign different colors (see the commands listed above) to arcs and edges, this is how the network map looks in Pajek.

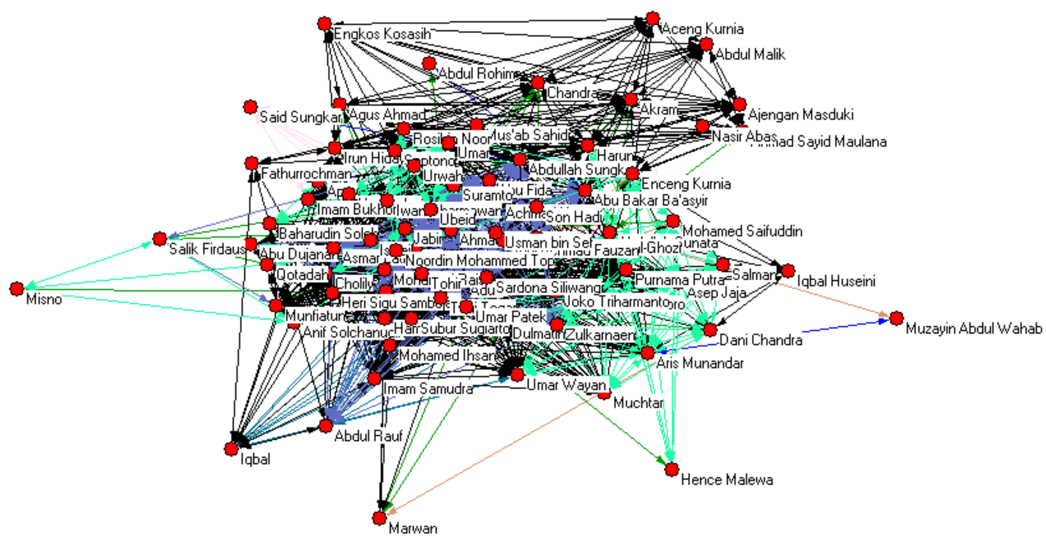


Figure 4.33: Pajek Multirelational Network Map of Noordin Network

Aggregating Data in Pajek

[UCINET] Data>Export >Pajek>Network

[Pajek] Network>Read

Nets>Union of Line

Draw>Draw

While we can aggregate data files in Pajek, unfortunately, we can only combine two files at a time, which is why you will probably do most of your file aggregation in UCINET. That said, we briefly illustrate how to aggregate files in Pajek. We will first need to export two of the Noordin files using UCINET's *Data>Export>Pajek>Network* command (not shown). Next, open Pajek and read the two files into memory using Pajek's *Network>Read* command. Remember to look for files with a .net extension because you exported the data from UCINET using the *Export>Pajek* option. To combine matrices in Pajek, you need to first identify the two matrices you intend to combine. In Pajek highlight the first matrix in the first network drop list and the second in the second network drop list (see Figure 4.34). Then, select the *Nets>Union of Lines* command. This will create a new network that Pajek labels *Fusion of 1 and 2*, which you can visualize the combined network using the *Draw>Draw* command.

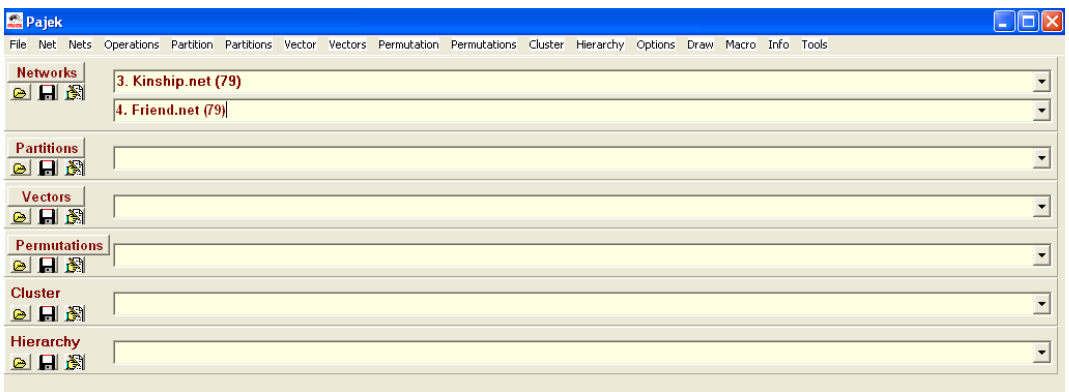


Figure 4.34: Two Networks Highlighted in Pajek's Network Dropdown Lists

Net>Transform>Remove >Multiple lines>Sum Values

When Pajek combines networks using the *Union of Lines* command, it treats them as a single relation with multiple lines. If you want to combine the two networks into a single valued network, make sure that the newly created (i.e., fused) network is highlighted in the first network drop list, then select the *Sum Values* command under the *Net>Transform>Remove>Multiple lines* submenu.

Positive and Negative Relations in UCINET and Pajek

In this section we will try something a bit different. As you know, the Sampson data contains both positive (e.g., like) and negative (dislike) sociometric data. We are going to multiply one of the negative networks ("dislike") by "-1" in order to transform the values into "negative" values and then add this newly created matrix to one of the positive networks ("like" at time 3). This combined

network will contain both positive and negative values in the various cells. We will then export this network to Pajek and visualize it.

[UCINET] Data>Display

Transform >Matrix Operations >Within dataset >Cellwise Transformations

First, confirm that the negative disliking relation (SAMPDLK.##h) is indeed coded with positive numbers by displaying this file in UCINET using the *Data>Display* command (or the display icon button). Next, we will transform these positive values into negative ones using the *Transform>Matrix Operations> Within dataset>Cellwise Transformations* command. This brings up the following dialog box (Figure 4.35). Note that we have selected “SAMPDLK.##h” as our input dataset and “SAMPDLK2.##h” as the name of the transformed (output) dataset. We have also checked the “Negative” box among the Transformation options. Click “OK” and you should get an output log that indicates that the values in the new matrix are negative.

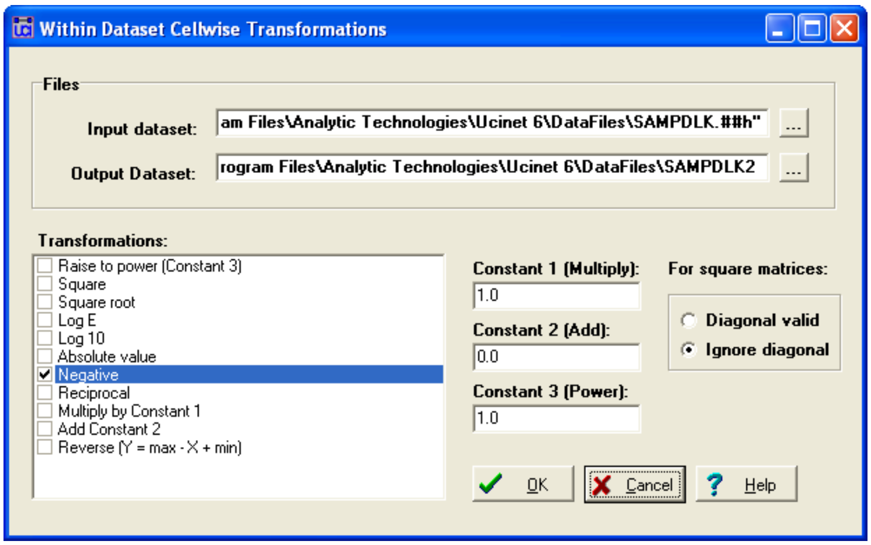


Figure 4.35: UCINET Within Dataset Cellwise Transformations Dialog Box

Transform >Matrix Operations >Between datasets >Statistical Summaries

The next step is to add this newly created matrix to the SAMPLK3.##h matrix. To do this, we will once again use UCINET’s *Transform>Matrix Operations>Between datasets>Statistical Summaries* command (see Figure 4.36, next page). After UCINET finished processing the command, an output log will appear that should show an aggregated matrix containing both positive and negative numbers (Figure 4.37, next page).

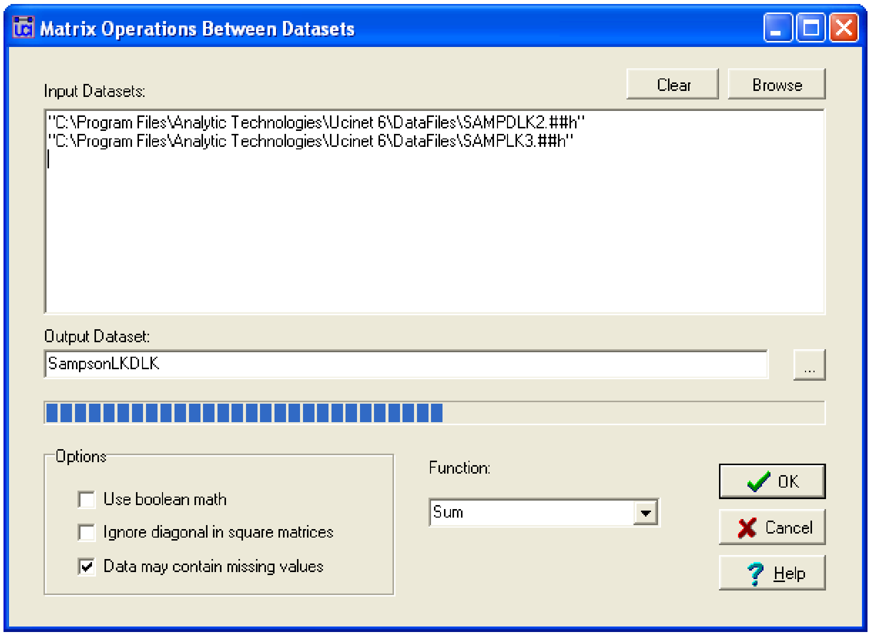
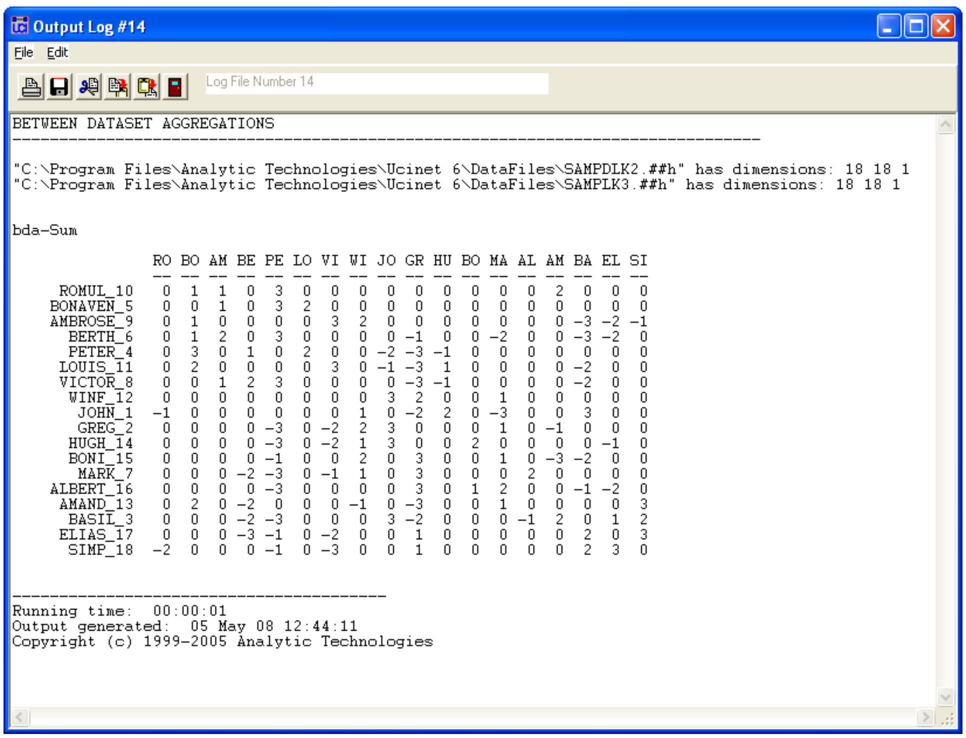


Figure 4.36: UCINET Between Dataset Statistical Summaries dialog box




```
BETWEEN DATASET AGGREGATIONS
------------------------------------------------------------------------------------

"C:\Program Files\Analytic Technologies\Ucinet 6\DataFiles\SAMPDLK2.##h" has dimensions: 18 18 1
"C:\Program Files\Analytic Technologies\Ucinet 6\DataFiles\SAMPLK3.##h" has dimensions: 18 18 1


bda-Sum

              RO BO AM BE PE LO VI WI JO GR HU BO MA AL AM BA EL SI
              -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
    ROMUL_10   0  1  1  0  3  0  0  0  0  0  0  0  0  0  2  0  0  0
   BONAVEN_5   0  0  1  0  3  2  0  0  0  0  0  0  0  0  0  0  0  0
   AMBROSE_9   0  1  0  0  0  0  3  2  0  0  0  0  0  0  0 -3 -2 -1
     BERTH_6   0  1  2  0  3  0  0  0  0 -1  0  0 -2  0  0 -3 -2  0
     PETER_4   0  3  0  1  0  2  0  0 -2 -3 -1  0  0  0  0  0  0  0
    LOUIS_11   0  2  0  0  0  0  3  0 -1 -3  1  0  0  0  0 -2  0  0
    VICTOR_8   0  0  1  2  3  0  0  0  0 -3 -1  0  0  0  0 -2  0  0
     WINF_12   0  0  0  0  0  0  0  0  3  2  0  0  1  0  0  0  0  0
      JOHN_1  -1  0  0  0  0  0  0  1  0 -2  2  0 -3  0  0  3  0  0
      GREG_2   0  0  0  0 -3  0 -2  2  3  0  0  0  1  0 -1  0  0  0
     HUGH_14   0  0  0  0 -3  0 -2  1  3  0  0  2  0  0  0  0 -1  0
     BONI_15   0  0  0  0 -1  0  0  2  0  3  0  0  1  0 -3 -2  0  0
      MARK_7   0  0  0 -2 -3  0 -1  1  0  3  0  0  0  2  0  0  0  0
   ALBERT_16   0  0  0  0 -3  0  0  0  0  3  0  1  2  0  0 -1 -2  0
    AMAND_13   0  2  0 -2  0  0  0 -1  0 -3  0  0  1  0  0  0  0  3
     BASIL_3   0  0  0 -2 -3  0  0  0  3 -2  0  0  0 -1  2  0  1  2
    ELIAS_17   0  0  0 -3 -1  0 -2  0  0  1  0  0  0  0  0  2  0  3
     SIMP_18  -2  0  0  0 -1  0 -3  0  0  1  0  0  0  0  0  2  3  0


----------------------------------------
Running time:  00:00:01
Output generated:  05 May 08 12:44:11
Copyright (c) 1999-2005 Analytic Technologies
```

Figure 4.37: UCINET Between Dataset Statistical Output Log

Data>Export>DL

Next, use the *Data>Export>DL* command (not shown) to export this aggregated matrix to Pajek. Remember to change the extension of the exported file to .dat. The reason for using this command to export the data rather than the

Data>Export>Pajek>Network command is because (at least currently) when you use the latter command, UCINET exports the negative cell values as positives.

Pajek

[Pajek] File>Network>Read

Draw>Draw

Layout>Energy >Kamada-Kawai>Free

Open Pajek and read in the network data you just exported from UCINET using the *File>Network>Read* command. (Remember that the network file you are trying to read has a .dat extension not a .net. one.) Next, visualize the network using the *Draw>Draw* command and energize it using one of Pajek's mapping algorithms (either Kamada-Kawai or Fruchterman Reingold or both). Note that there are both solid lines and negative lines between the nodes. In Pajek solid lines indicate positive connections while dotted lines indicate negative connections.

Options>Size>of Vertices

If the nodes are hard to distinguish with all of the lines, you can increase the size of the node using the *Options>Size>of Vertices* command in Pajek's Draw screen. Figure 4.38 uses a vertex size of 10. Generally, however, it is wise to use the default setting ("0"), which tells Pajek to automatically set the size of the vertices. This is an especially useful option when node size reflects an attribute that varies considerably (e.g., age, centrality, etc.).

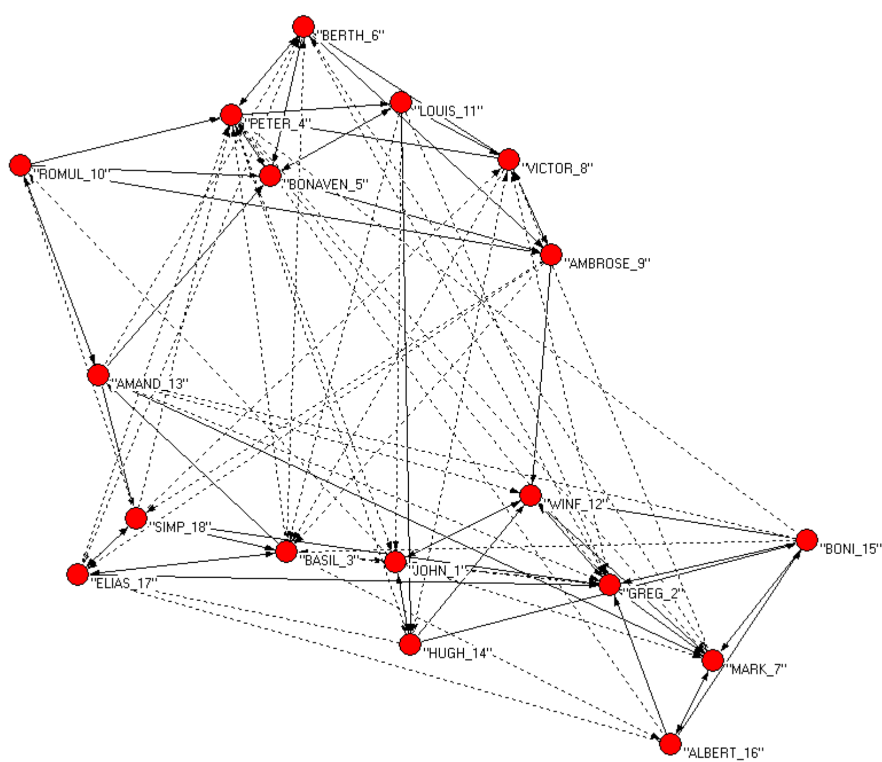


Figure 4.38: Pajek Drawing of Sampson Monastery Liking and Disliking Data

4.7 Summary and Conclusion

This chapter has focused on methods for collecting and recording social network data. It began with a brief discussion on how social network analysts define the boundaries of the networks they are examining. It then moved to a brief discussion of the difference between ego and complete networks, noting that social network analysts generally use complete networks when applying social network methods. Indeed, most social network methods assume that one is working with complete social network data. Next, we considered the different types of data social network analysts use: one-mode network data, two-mode network data and attribute data. We then discussed the various ways that social network analysts collect network data before looking at how social network data are recorded in matrix form. The final section of this chapter then looked at various ways of aggregating multiple networks into both "stacked" and single-value matrices. It is now time to turn our attention to the various families of metrics that analysts use to examine social networks.

Chapter 5

Network Topography

Networks can differ considerably from one another, and there is considerable evidence that suggests that a network's topography (i.e., its overall structure) has an effect on its performance and/or efficiency. For instance, social movement theorists have found that two measures of a network's overall characteristics – network density and network centralization – impact a social movement's effectiveness (Diani 2002; Osa 2003), and Ami Pedahzur and Arie Perliger (2006) have noted that terrorist networks with a large number of cliques (see Chapter 7) appear to be more effective than those with fewer. If network topography does have an impact on its effectiveness, then examining the topography of dark networks may provide us with hints as to which strategies we should adopt for rendering them less effective. For instance, earlier we saw that the global salafi jihad exhibits the characteristics of a scale-free network (Sageman 2004b), which has led some to argue that we should focus our efforts on targeting hubs (i.e., actors with numerous ties) rather than randomly stopping terrorists at our borders (Sageman 2004a). While others have questioned the utility of this approach (Pedahzur and Perliger 2006; Tsvetovat and Carley 2005), it does illustrate how a dark network's overall structure can suggest possible strategies for disruption. To get us thinking about network topography, we begin by briefly returning to Granovetter's discussion of weak and strong ties.

5.1 Weak Ties and Strong Ties Redux

Earlier we saw how Mark Granovetter's (1973; 1974) discovered that when it came to finding jobs, people were far more likely to use personal contacts than other means. Moreover, of those who found their jobs through personal contacts, most of the personal contacts were weak ties (i.e., acquaintances) rather than strong ties (i.e., close friends). This was because not only do we tend to have more weak ties and strong ties (because weak ties demand less of our time), but also because our weak ties are more likely to form the crucial bridges that tie together densely knit clusters of people (see Figure 5.1 below). In fact, if it were not for these weak ties, Granovetter argues, these clusters would not be connected at all. Thus, whatever is to be spread (e.g., information, influence, and other types of resources), it will reach a greater number of people when it passes through weak ties rather than strong ones. Moreover, Granovetter argues that actors whose

with few weak ties are more likely to "confined to the provincial news and views of their close friends" (Granovetter 1973).

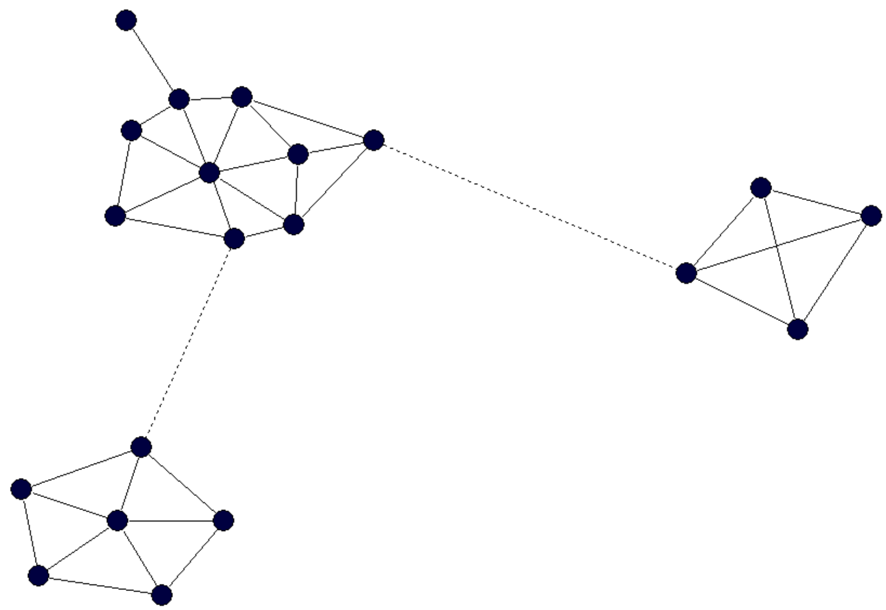


Figure 5.1: Strong and Weak Ties

Granovetter does not argue that strong ties were of no use, however. He notes that while weak ties provide individuals with access to information and resources beyond those available in their immediate social circles, strong ties have greater motivation to be sources of support in times of uncertainty. Others have noted this as well. "There is a mountain of research showing that people with strong ties are happier and even healthier because in such networks members provide one another with strong emotional and material support in times of grief or trouble and someone with whom to share life's joys and triumphs" (Stark 2007:37).

Implicit in Granovetter's argument is that people's networks differ in the number of weak and strong ties they have. Their networks can range from local or provincial ones consisting of primarily of strong, redundant ties and very few weak ties to worldly or cosmopolitan networks consisting of numerous weak ties and very few strong ties (Stark 2007:37-38). Moreover his analysis implicitly suggests that that ideally peoples' networks should consist of a mix of weak and strong ties. Indeed, as we shall see below (Section 5.2) Pescosolido and Georgianna's study of suicide suggests that individuals should seek to embed themselves within a network characterized by a mix of weak and strong ties: weak ties to provide them access to the flow of information and other resources

throughout the social structure and strong ties to provide them with support and a strong sense of belonging, meaning and identity.

Weak and Strong Ties in the New York Apparel Industry

An ideal mix of weak and strong ties appears to not only provide benefits at the individual level but also at the organizational level. In his study of the New York apparel industry, Brian Uzzi (1996) found that a mix of weak and strong ties proved beneficial to the long-term survival of apparel firms.[20] The firms he studied tended to divide their market interactions into two types: "market" or "arms-length" relationships (i.e., weak ties) and "special" or "close" relationships (i.e., strong ties), which Uzzi refers to as "embedded" ties. He found that while market relationships were more common than embedded ties, they tended to be less important. Embedded ties were especially important in situations where trust was paramount, where fine-grained information had to be passed to the other party, and when certain types of joint problem-solving were on the agenda (Uzzi 1996:677).

According to Uzzi, embeddedness increases economic effectiveness along a number of dimensions crucial to competitiveness in the global economy: organizational learning, risk-sharing and speed-to-market. However, he also found that firms that are too embedded suffer because they cease to have access to information from distant parts of the network, which makes them vulnerable to rapidly changing situations. This lead Uzzi to argue that firms should seek to maintain a balance of embedded and market ties. In support of this he found that the topography of interfirm networks (i.e., in terms of embedded and market ties) varied and that a U-shaped or curvilinear relationship exists between the degree of embeddedness and the probability of firm failure (Uzzi 1996:675-676). Firms that exhibited extremely high levels of embedded ties (i.e., provincial networks) or market ties were much more likely to fail than those that maintained a balance between the two.

5.2 Network Density, Average Degree and Centralization

One measure that attempts to get at a network's mix of strong and weak ties is network density. Density is formally defined as the total number of ties within a network divided by the total possible number of ties, which means that network density measures range from 0.0 to 1.0. In "networks" with a density of 0.0, no ties exist between actors, while in networks with a density of 1.0 all possible ties

[20] Uzzi does not use the weak and strong tie terminology in the article.

exist between actors. As noted in Chapter 2, researchers have found that network density is positively related to the likelihood that actors within the network will follow accepted norms and behavior (Granovetter 2005). Another type of behavior that is apparently affected by network density is suicide. Bernice Pescosolido and Sharon Georgianna (1989) reframed Durkheim's (1951) classic study of suicide[21] in terms of social network theory, arguing that social network density has a curvilinear relationship to suicide. According to Pescosolido and Georgianna, individuals who are embedded in very sparse and very dense social networks are far more likely to commit suicide than are people who are embedded in moderately dense networks. Why? People who exist in sparse social networks often lack the social and emotional ties that provide the support that people often need during times of crisis. Moreover, they lack the social ties that might otherwise prevent them from engaging in self-destructive (i.e., deviant) behavior.[22] On the other hand, people who are involved in highly dense networks are often cut-off from people outside of their immediate social group. Thus, they often lack the ties to people (i.e., friends, family) who might stop them from taking the final, fatal step. Moreover, since it is common in highly-clustered groups for the group's opinion to shift toward extreme versions of their commonly-held beliefs, it is not surprising that sometimes these groups turn to extreme behaviors such as suicide. The Jonestown and Heaven's Gate mass suicides appear to be an example of this.

Network Density in UCINET and Pajek

[UCINET] Network>Cohesion >Density >(new) Density Overall

Transform>Dichotomize

To see how to calculate network density in UCINET and Pajek, we will use the "Noordin Combined Network" file, which is a stacked matrix of ten (10) different Noordin (one-mode) networks. In UCINET you calculate network density using the *(new)Density Overall* command found under the *Network> Cohesion>Density* submenu. Before doing this, however, we need to first dichotomize (binarize) the network because some of the networks in this file are valued (e.g., two people may have two or more ties because they participated in two or more of the same training events or belong to two more of the same organizations). To dichotomize a network in UCINET, select the *Transform> Dichotomize* command, which calls up a dialog box (Figure 5.2 below) that asks

[21] Durkheim essentially that social integration and social regulation have a curvilinear effect on suicide. High levels of integration and regulation are positively related to the suicide rate as are low levels of integration and regulation. By contrast moderate levels of integration and regulation are negatively related to the suicide rate.

[22] Recall that people who live in dense networks are more likely to follow accepted norms and behavior.

you to indicate which data file you want to dichotomize. Tell UCINET to use the "Noordin Combined Network" file. Note also that UCINET also provides a "cut-off operator" and "cut-off value" options that allow you to tell UCINET what you want your cutoff value for dichotomizing the network to be. Here, we have chosen to accept UCINET's default because we want every value in each of the ten matrices that is greater than zero to be coded to 1. Not also that has a default setting for the output file name.

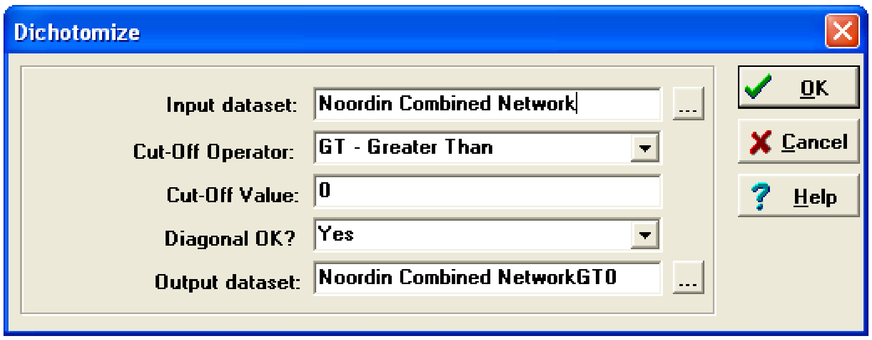


Figure 5.2: UCINET's Dichotomize Dialog Box

[UCINET] Network>Cohesion >Density >(new) Density Overall

Now, we are ready to calculate the density of the ten networks. UCINET's new overall density command calls up a dialog box (see Figure 5.3 below) where you should indicate as your input dataset the "Noordin Combined NetworkGT0" network (i.e., the recently dichotomized network). Click OK. The resulting output log will give you density scores for each of the networks in the file.

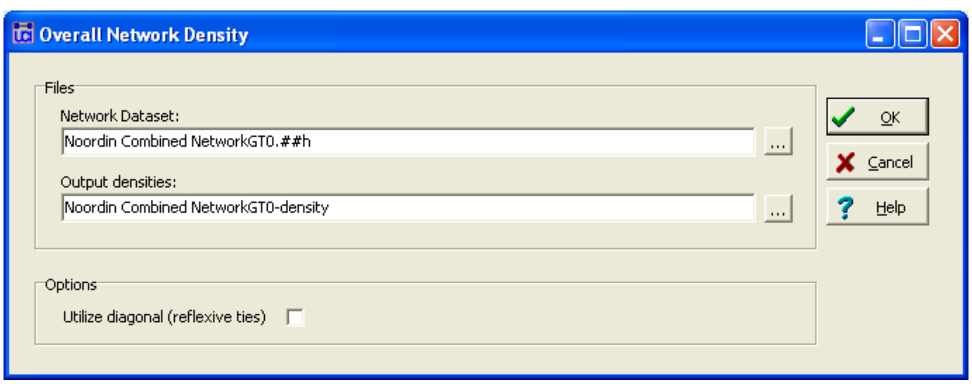


Figure 5.3: UCINET's Density Dialog Box

Transform >Matrix Operations >Within dataset >Aggregations

Transform>Dichotomize

Network>Cohesion >Density >(new) Density Overall

Of course, we may also be interested in aggregating the ten networks into a single network and then calculating the resulting network's density. To do this we use UCINET's *Transform>Matrix Operations>Within datasets>Aggregations* command, which we discussed at length in Chapter 4. After aggregating the stacked matrix, we then need to dichotomize the network (because now some cells have values of greater than one) using UCINET's *Transform>Dichotomize* command. After aggregated matrix is dichotomized, then we would calculate the density of the overall network using the *Network>Cohesion>Density>(new) Density Overall* command.

While calculating density for individual networks is very simple in Pajek, it is somewhat more complicated to calculate density for multirelational networks, which is why when working with a stacked matrix, you will probably want to estimate your density measures in UCINET. Nevertheless, it is helpful to see how it is done so you do not need to switch back and forth between programs unnecessarily.

[UCINET] Data>Export>DL

[Pajek] Net>Transform >Arcs→Edges>All

Net>Transform >Remove Loops

Net>Transform >Multiple Relations >Extract Relations

Info>Network>General

Before calculating network density in Pajek we need to first export the "Noordin Combined NetworkGT0" file from UCINET to Pajek using UCINET's *Data>Export>DL* command so that the various types of relations will be preserved in Pajek. After importing the resulting .dat file into Pajek, saving it as a .net file, removing any double quotes so that labels can be seen in Pajek network maps, and (finally) assigning for labels for each of the type of relations (see Chapter 4 above), we need to symmetrize and remove any loops from the network before we calculate network density. To do this we use Pajek's *Net>Transform> Arcs→Edges>All* command. This brings up a dialog box asking you whether you want to create a new network. Select "Yes." Pajek will then ask whether you want to remove multiple lines. Choose the "Single" option (#5). Next, select the *Net>Transform>Remove Loops* command. (We do this so that the diagonal is not used in calculating the network's density.) Once again you will be asked to create a new network; once again select yes.[23] Finally, we need to extract each relation before we can calculate their density. To do this we use Pajek's *Net>Transform> Multiple Relations>Extract Relation(s)* command. In the resulting dialog box tell Pajek that you want relations 1-10. To estimate the density of a network, make sure that it is showing in the network dropdown list and select the *Info>Network> General* command. This will call up Pajek's report window which displays network density for that network (not all of the networks) with and without loops (i.e., including the cells along the diagonal into the calculation and not). In other

[23] You will probably want to rename the newly-created networks in order to keep all the networks in the project file straight.

words: you cannot simultaneously estimate the density of all the networks. Pajek's results (i.e., the without loops calculation) should agree with UCINET's.

Average Degree in UCINET and Pajek

Unfortunately, network density scores tend to decrease as social networks get larger because the number of possible lines increases rapidly with the number of actors whereas the number of relations which each actor can maintain is generally limited (at least for individuals – corporations and groups may display a different dynamic). Consequently, network density is of limited value. We can use it to compare networks of the same size, but that is about all. An alternative suggested by Scott (2000:75-76) and de Nooy et al (2005:63) is to calculate a network's average degree centrality. Not only does average degree centrality tend to be positively associated with denser networks, but unlike network density, it is not sensitive to network size, which means that we can use it to compare networks with different numbers of nodes. While we will discuss various centrality measures in more depth in the next chapter, here we will note that degree centrality is the number of ties (neighbors) that each actor has in a network. Average degree centrality, then, is the average of every individual actor's degree centrality score. To calculate average degree centrality in UCINET, use its *Network>Centrality>Degree* command; this generates an output log containing centrality scores for individual actors. Scroll past the individual centrality scores for each matrix, and you will find an average (mean) centrality score (Figure 5.4).

[UCINET] Network>Centrality >Degree

```
OUTPUT.LOG1 - Notepad
File  Edit  Format  View  Help
67     Sardona Siliwangi      0.000       0.000       0.000
68              Son Hadi      5.000       6.410       0.147
69        Subur Sugiarto      0.000       0.000       0.000
70               Suramto      3.000       3.846       0.088
71                 Tohir      0.000       0.000       0.000
72            Toni Togar      0.000       0.000       0.000
73                 Ubeid      0.000       0.000       0.000
74                  Umar      0.000       0.000       0.000
75            Umar Patek      0.000       0.000       0.000
76            Umar Wayan      0.000       0.000       0.000
77                 Urwah      0.000       0.000       0.000
78         Usman bin Sef      3.000       3.846       0.088
79            Zulkarnaen      0.000       0.000       0.000


DESCRIPTIVE STATISTICS

                            1            2            3
                       Degree    NrmDegree        Share
                 ------------ ------------ ------------
 1     Mean           0.430        0.552        0.013
 2  Std Dev           1.122        1.438        0.033
 3      Sum          34.000       43.590        1.000
 4 Variance           1.258        2.067        0.001
 5      SSQ         114.000      187.377        0.099
 6    MCSSQ          99.367      163.325        0.086
 7 Euc Norm          10.677       13.689        0.314
 8  Minimum           0.000        0.000        0.000
 9  Maximum           6.000        7.692        0.176


Network Centralization = 7.33%
Heterogeneity = 9.86%.  Normalized = 8.71%
```

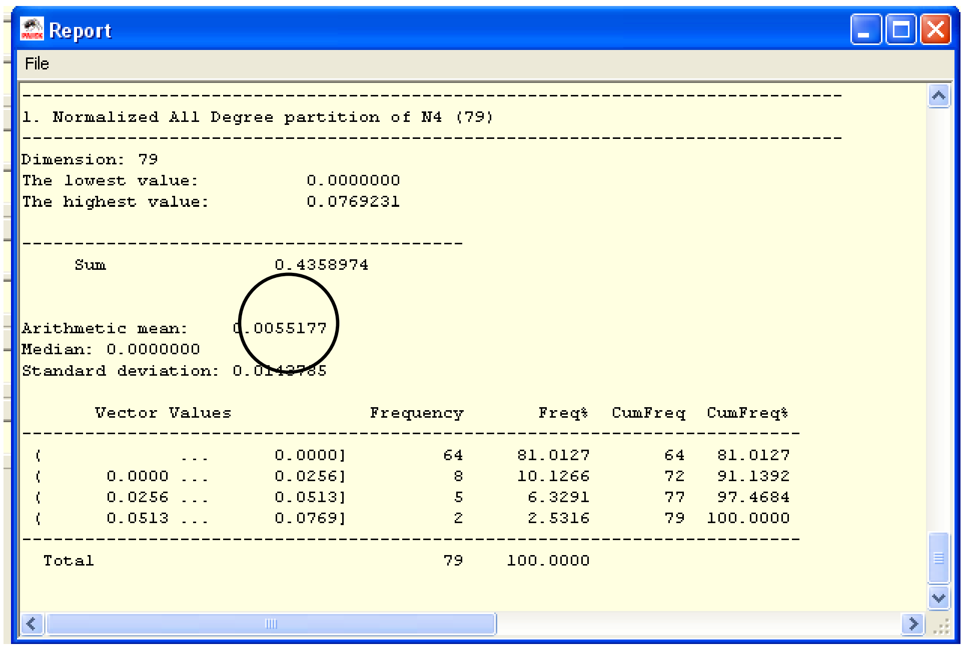
Figure 5.4: UCINET's Output Log Indicating Average (Mean) Degree Centrality

As with Pajek's approach to estimating network density, you have to calculate average degree centrality one network at a time. This is not a drawback when you are only working with one network, but when working with multirelational data, it makes it somewhat cumbersome.

[Pajek] Net>Partitions>Degree>All

Info>Vector

To calculate average degree centrality in Pajek for each network, use the *Net>Partitions>Degree>All* command. This command generates both a partition and a vector. To get a network's *normalized* average degree centrality, select the *Info>Vector* command (accept Pajek's defaults). This brings up Pajek's report window (see Figure 5.5 below), which lists the *normalized* network's average degree centrality (arithmetic mean).




```
-----------------------------------------------------------------------------
1. Normalized All Degree partition of N4 (79)
-----------------------------------------------------------------------------
Dimension: 79
The lowest value:          0.0000000
The highest value:         0.0769231

------------------------------------------
     Sum                0.4358974


Arithmetic mean:     0.0055177
Median: 0.0000000
Standard deviation: 0.0143785

       Vector Values              Frequency       Freq%  CumFreq  CumFreq%
-------------------------------------------------------------------------
 (             ...      0.0000]          64     81.0127       64   81.0127
 (      0.0000 ...      0.0256]           8     10.1266       72   91.1392
 (      0.0256 ...      0.0513]           5      6.3291       77   97.4684
 (      0.0513 ...      0.0769]           2      2.5316       79  100.0000
-------------------------------------------------------------------------
  Total                                  79    100.0000
```

Figure 5.5: Pajek's Report Window

Partition>Make Vector

Upon closer inspection, you will notice that normalized average degree centrality is the same as network's density. In order to get average degree centrality, we need to first create a new vector with the partition that was generated when we first calculated degree centrality above. With that partition highlighted in the partition drop box, select Pajek's *Partition>Make Vector* command. With the newly created vector, choose Pajek's *Info>Vector* command, accepting Pajek's defaults and click OK. Pajek's report window will appear (Figure 5.6, next page) with average degree centrality reported (arithmetic mean).

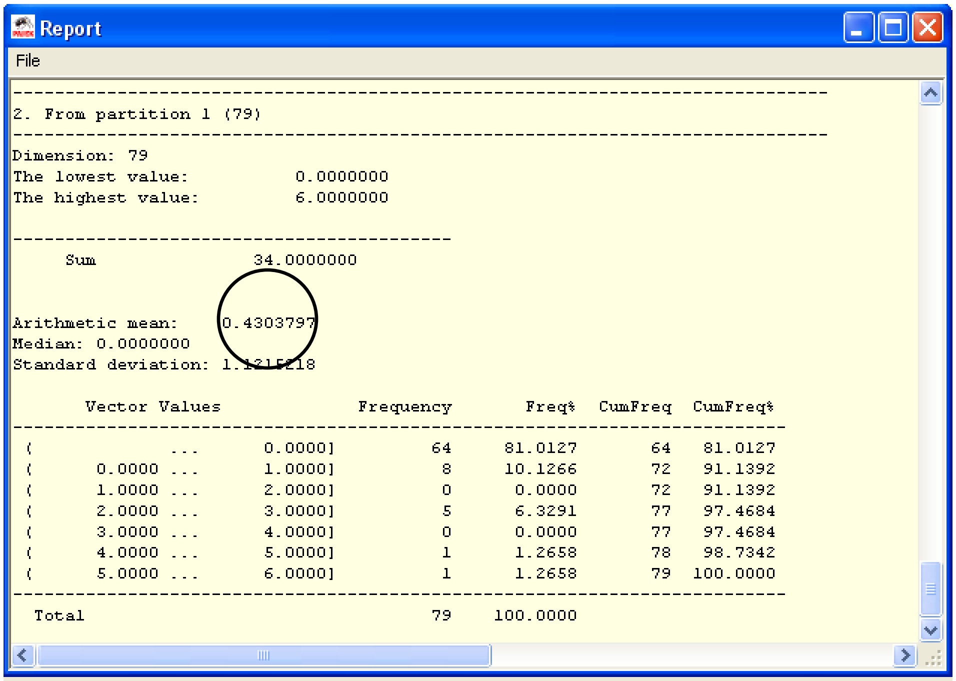


Figure 5.6: Pajek's Report Window

Network Centralization in UCINET and Pajek

[UCINET]
Centrality>Degree

[Pajek]
Net>Partitions>Degree
>All, Input, Output

Network centralization is another measure related to degree centrality that attempts to capture the overall makeup of a network. It helps determine how centralized (e.g., hierarchical) or decentralized (e.g. "flat" organizations) a network is. The algorithm focuses on the variation in actor centrality within the network to measure the level of centralization. More variation yields higher network centralization scores, while less variation yields lower scores. In UCINET network centralization scores are generated when you calculate degree centrality. If you look again at Figure 5.6 above, you will note that the network centralization score appears just below the descriptive statistics. Pajek, like UCINET, generates centralization scores when it calculates centrality and appears in the report window (not shown) that is called up when we issue the command. Centralization scores range from 0 – 1 (or 0 – 100%) when you are working with dichotomized data. If you are working with valued data, centralization scores will sometimes be larger than one (and generally meaningless). Thus, you will want to dichotomize your data before estimating network centralization.

Table 5.1 presents the density, average degree and centralization scores for each of the 10 networks as well as scores for the overall network. Rankings are in parentheses. As you can see density and average degree correlate with one another. In fact, in this case, since the networks are of the same size, they

correlate perfectly with one another. Network centralization is another matter, however. While the organizational network is clearly the most dense (whether measured by density or average degree), it is not the most centralized. Instead internal communications is, which illustrates why we should always consider multiple measures when examining network topography.

Type of Tie	Density	Average Degree	Centralization
Business & Financial	.0055 (8)	.430 (8)	7.33% (8)
Friendship	.0286 (6)	2.228 (6)	12.85% (7)
Internal Communications	.0552 (2)	4.304 (2)	41.69% (1)
Kinship	.0039 (9)	.304 (9)	2.23% (10)
Logistical Place	.0166 (7)	1.291 (7)	15.40% (6)
Operational	.0305 (5)	2.380 (5)	23.18% (4)
Organizational	.3158 (1)	24.633 (1)	28.11% (2)
Religious	.0039 (9)	.304 (9)	4.86% (9)
School	.0552 (2)	4.304 (2)	23.28% (3)
Training	.0396 (4)	3.089 (4)	15.67% (5)
Overall	.3775	29.433	40.19%

Table 5.1: Comparison of Density, Average Degree and Centralization Scores (Rank in Parentheses)

5.3 Weak Ties and Small Worlds

Recent research that implicitly builds upon Granovetter's (1973; 1974) notion of weak and strong ties and Duncan Watts's (Watts 1999a, 1999b, 2003; Watts, Dodds and Newman 2003) analysis of small worlds has yielded alternative network topography measures that appear to explain network performance. We begin by considering a research project of Brian Uzzi and Jarrett Spiro that applies similar distinctions to the network of creative teams that produced Broadway musicals in the latter half of the 20th century and finds that variations in the small

world properties of the yearly networks of these creative teams helped to predict whether musicals would be financial and critical successes. This gives way to an overview of how to estimate the relevant measures in UCINET and Pajek.

Weak Ties, Small Worlds and Network Performance

Uzzi and Spiro (2005) explored the effects that network topography had on the success of the creative teams that created Broadway musicals from 1945 to 1989.[24] Using measures developed by Duncan Watts and his colleagues (Watts 1999a, 1999b, 2003; Watts, Dodds and Newman 2003), Uzzi and Spiro determined the extent to which each network of creative teams (as determined by year) exhibited small world characteristics. According to Watts a network exhibits small world characteristics when (compared to random networks of the same size) actors cluster into tight-knit groups (as measured by the average clustering coefficient – see below) and the average path length between them (as measured by path distance) is low. The more that the clustering coefficient ratio (clustering coefficient of the actual network/clustering coefficient of the random network) exceeds 1.0 and the closer the path length ratio (path length of the actual network/path length of the random network) approaches 1.0, the more the network acts like a small world. Uzzi and Spiro discovered that the ratio of these two measures (the clustering coefficient ratio divided by the path length ratio – which they called the small world quotient or small world Q) has a curvilinear relationship with the probability that a musical would be a critical and financial success.[25]

Why do they believe this curvilinear relationship between global clustering and Broadway success exists? He argues that up to a point connectivity and cohesion between members of the various teams that produced the musicals is probably beneficial because it "increases their access to diverse and novel creative material circulating in all parts of the small world" (Uzzi 2008:9). However, as connectivity increases it eventually reaches a point where returns turn negative. "Very high levels of connectivity and cohesion may lead to a homogenization and imitation of the same ideas by the different teams in the network, lowering the opportunity for individual teams to distinguish themselves with an exceptional show material" (Uzzi 2008:9). In other words up to a point, connectivity increases a network's overall creativity by encouraging human innovation, but beyond that, connectivity actually appears to stifle it.

---

[24] We will also consider the results of a follow up study conducted just by Uzzi (2008).
[25] Later analysis by Uzzi (2008) found that it was unnecessary to compute the small world quotient to predict the probability that a musical would be a critical and financial success. All that was needed was the clustering coefficient ratio since the path length ratio tended to remain around 1.0.

Estimating Clustering Coefficients and Path Distance in UCINET

[UCINET] Network>Cohesion >Clustering Coefficient

Estimating global clustering coefficients in UCINET and Pajek are relatively simple. The tricky part is to remember to not only estimate the clustering coefficient for the network under examination but also *the clustering coefficient for a random network of the same size*. In UCINET select the *Clustering Coefficient* command located under the *Network>Cohesion* submenu, which calls up the following dialog box (Figure 5.7).

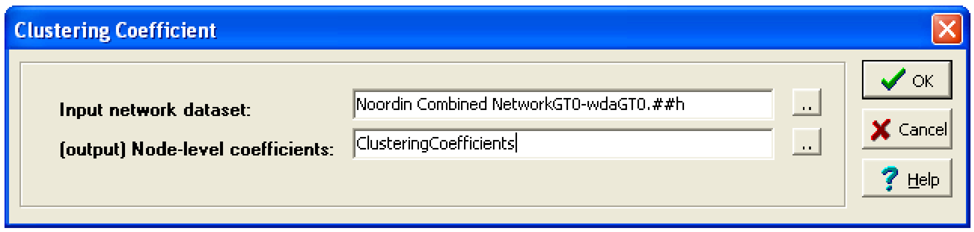


Figure 5.7: UCINET Clustering Coefficient Dialog Box

Here we have selected the network we aggregated and dichotomized from the original stacked matrix (i.e., the overall network in Table 5.1) as our input network dataset. Click OK and UCINET will generate an output file (not shown) that first lists the "overall graph clustering coefficient" (in this case, .773). UCINET calculates the global clustering coefficient by first estimating the clustering coefficient for each actor in the network, summing these together and then dividing the result by the number of actors that are not isolates (note that some actors do not have a clustering coefficient score). This method differs from Pajek's approach, which divides the total score by the total number of actors in the network, regardless of whether some have a clustering score or not. The next step is to compare this clustering coefficient with that of a random network of the same size. To do this select the *Erdos-Renyi random graph* command found under the *Data>Random* submenu, which brings up a following dialog box (Figure 5.8):

[UCINET] Data>Random >Erdos-Renyi random graph

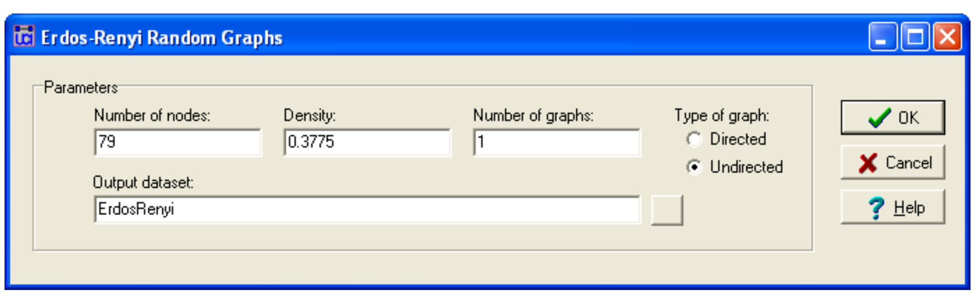


Figure 5.8: UCINET's Random Graph Dialog Box

Note that in order to create a random network you need to tell UCINET how many nodes (actors) are in the network (in this case 79) and the density of the

network. Here, use the same density of the actual network, which means that you will first need to compute that (see Table 5.1 above) before performing this operation. Once you have provided the necessary information, click OK, and then compute the clustering coefficient of this newly-created graph (in this case it = .386). The resulting ratio thus equals .773/.386 = 2.003.

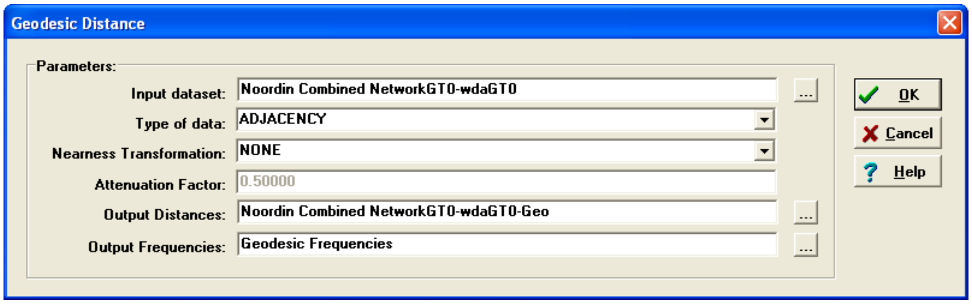


Figure 5.9:   UCINET's Geodesic Distance Dialog Box

Network>Cohesion >Distance

UCINET calculates path distance with its *Network>Cohesion>Distance* command, which brings up a dialog box similar to Figure 5.9. Of course we need to calculate the average path distance for both the original network (1.644) and the random network (1.623), before calculating the ratio of the two measures (1.013).

Estimating Clustering Coefficients in Pajek

[Pajek] Net>Vector >Clustering Coefficients >CC1

File>Vector>Edit

Info>Vector

To calculate the clustering coefficient in Pajek, export the network data we just used in UCINET to Pajek and read it into Pajek's memory. Next, select the *Net>Vector>Clustering Coefficients>CC1* command. This creates a partition and two vectors. An examination of the CC1 vector ("not the CC1' vector) using Pajek's *File>Vector>Edit* command shows that Pajek has computed clustering coefficients for each actor in the network and that they equal those calculated by UCINET, except that Pajek assigns clustering coefficient scores of 0.000 to isolated actors whereas UCINET just leaves them blank. To get the network's overall clustering coefficient, make sure that "Clustering Coefficients CC1 in N1" is highlighted in the vector dropdown list, and then select *Info>Vector* command. This calls up two dialog boxes. Accept Pajek's defaults for both, click OK, and Pajek generates a report that looks like Figure 5.10 (next page). Pajek's calculation of the clustering coefficient (.7629) differs from UCINET's because Pajek divides the sum of the individual clustering coefficient scores by the total number of actors in the network, not the number of non-isolate actors as UCINET does. Indeed, if you take the sum of the individual clustering coefficients, which is listed in the report window (60.27) and divide it by the total number of non-isolate actors (78), you will arrive at the same score as did in UCINET.

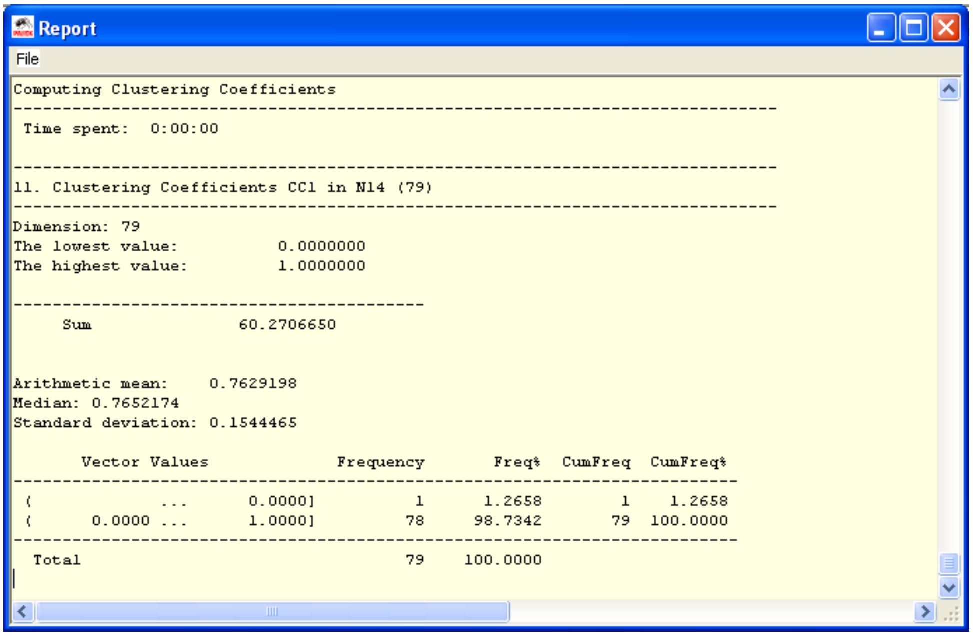


Figure 5.10: Pajek's Report Window with Clustering Coefficient Data

Net>Path between 2 vertices >Distribution of Distances >From All Vertices

Path distance is calculated in Pajek using Pajek's *Net>Path between 2 vertices>Distribution of Distances>From All Vertices* command. This brings up the following report window. As you can see the average distance as calculated by Pajek (1.644) is the same as that calculated by UCINET (Figure 5.11).

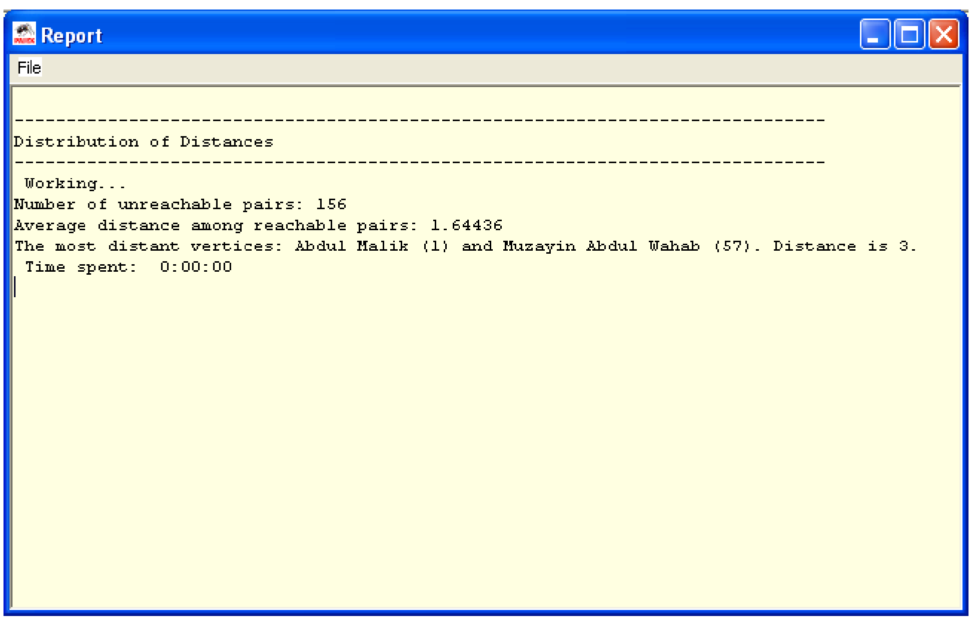


Figure 5.11: Pajek's Report Window with Path Distance Data

Net>Random Network >Erdos-Renyi>Undirected >General

To create a random graph in Pajek we use the *Net>Random Network> Erdos-Renyi>Undirected>General* command. This brings up two dialog boxes (not shown). The first asks how many vertices will be in the network (in this case 79). The second asks for the average degree centrality of the network you are creating. You want to use the same average degree as in the actual network (see Table 5.1). Pajek will generate a random network, for which you can then estimate a clustering coefficient (.3619) and average path distance (1.642) using the commands discussed above. Dividing the clustering coefficient of the original network by the clustering coefficient of the random network yields a clustering coefficient ratio of 2.108, which is slightly higher than calculated by UCINET, and a path distance ratio of 1.001, which is slightly lower than calculated by UCINET.

Info>Vector

Small World Q and Dark Networks

Table 5.2 summarizes UCINET's and Pajek's calculations as well as calculating Uzzi and Spiro's small world quotient. As you can see the calculations differ slightly from one another. This is due to two factors: (1) UCINET and Pajek use slightly different assumptions in generating random graphs, and as we have already seen (2) they calculate the overall clustering coefficient for networks somewhat differently. This raises an important question. Which program should we use to estimate the clustering coefficients, average path distances and small world quotients? Either one will probably do as long as we consistently use the same program. If we jump back and forth, however, we could draw incorrect conclusions about the networks we are tracking and comparing.

	UCINET	Pajek
Clustering Coefficient Ratio	2.003	2.108
Average Path Distance Ratio	1.013	1.001
Small World Quotient	1.977	2.106

Table 5.2: Comparison of UCINET and Pajek calculations of Clustering Coefficient Ratio, Average Path Distance Ratio and Small World Q

In spite of their differences, both programs yield a small world quotient of approximately 2.00. Is this large or small? At this point, we really do not know,

and we will not know until numerous case studies of dark networks are conducted. What is more, in carrying out these case studies we need to be careful that we compare similarly structured networks. Here we have used a network composed of ten (10) different types of ties for calculating density, average degree centrality, centralization and small world quotients, whereas other studies have only focused on friendship, kinship and mentor ties formed at school and/or through religious affiliations (Pedahzur and Perliger 2006; Sageman 2004b). In order for researchers to know whether relative to other networks a network is provincial or cosmopolitan, we will have to be sure that we are comparing networks constituted by the same types of ties.

5.4 Summary

While it may be (morally) difficult for some of us to think of dark networks as varying in their ability to encourage innovative thinking, the studies mentioned in this chapter should give us pause. They suggest that in order to be successful dark networks can be neither too provincial nor too cosmopolitan. Instead, they must maintain a mix of weak and strong ties (Figure 5.11 below). These studies also suggest that strategically we may want to promote policies that "push" dark networks toward being either too provincial or too cosmopolitan, that is, to high or low levels of global clustering.

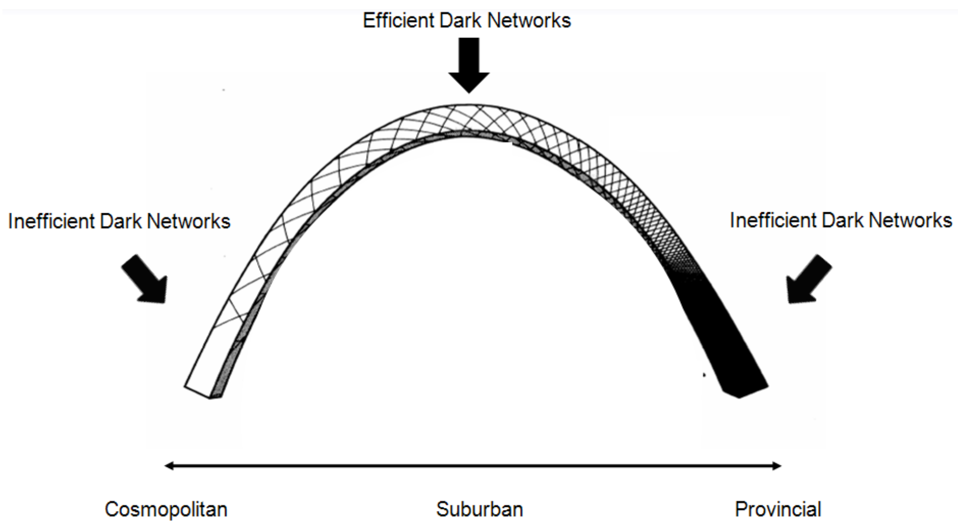


Figure 5.11: Relationship Between Network Topography and Network Efficiency (Note: Figure adapted from Pescosolido and Georgianna, Figure 2 (1989:44)

Take, for example, a scenario where the dark network we are seeking to disrupt exists on the "right" side of the "cosmopolitan-provincial" continuum. If in such a scenario we targeted a central actor within the network for capture or elimination, then that could have the effect of making the network less provincial (i.e., more cosmopolitan) and more effective. Instead, we may want adopt a strategy (or strategies) that cause the network to turn in on itself (e.g., peeling off peripheral members) making it more provincial (and thus less effective). An important thing to keep in mind as we move forward to examine other social network metrics, is that *no two networks are exactly alike*, and a network's overall characteristics will most likely impact its performance and efficiency. Moreover, how a network is structured should provide broad guidelines for choosing strategies for rendering it less effective.

CHAPTER 6

CENTRALITY, POWER AND PRESTIGE

Centrality is one of the oldest concepts in social network analysis. Notions that certain actors are more central than others go back at least as far as Jacob Moreno's conception of sociometric stars and isolates (Moreno 1953). And according to Scott (2000), Alex Bavelas and Harold Leavitt were the first to formally investigate the formal properties of centrality as they looked at how a network influences the flow of communication in experimental groups. These experiments usually involved the artificial partitioning of groups in various ways such that messages could only flow in certain directions and through particular persons. Not surprisingly, when communications had to flow through centrally located actors, certain patterns of behavior emerged, and when they did not or were altered, different patterns emerged.

Another contribution to the development of notions of network centrality comes from the exchange theory associated with Richard Emerson and Karen Cook (Cook and Emerson 1978; Cook et al. 1983; Cook, Gillmore and Yamagishi 1986; Cook and Whitmeyer 1992; Emerson 1962, 1972a, 1972b, 1976). Interestingly, though, this theoretical tradition is seldom mentioned in social network texts even though some of their experiments test the effects of centrality on power, which is of central concern among social network analysts. This is probably because of exchange theory's close association with rational choice theory, which not only assumes that actors are utility maximizing individuals but takes issue with certain structuralist positions in sociology that hold that all important social phenomena can be explained, if not completely, at least substantially by social structure (see Chapter 2 above). However, exchange theory does not discount the importance of networks. Like social network analysis it conceptualizes social structure as the configuration of social relations and positions, which is why it is sometimes referred to as exchange network theory (or network exchange theory). Moreover, the fundamental unit of analysis in exchange theory is not the autonomous actor but rather the exchange relationship between actors. Nevertheless, exchange theory does depart from those forms of network analysis that focus solely on social structure and do not take the interests of individual actors into consideration (Cook and Whitmeyer 1992).

Researchers conceptualize centrality in a variety of ways (Bonacich 1987; Freeman 1979; Wasserman and Faust 1994). A central actor can be seen as someone who has a lot of ties to other actors (degree centrality), as someone who

is close (in terms of path distance) to all other actors than are others in the network (closeness centrality), as someone who lies on the shortest path between numerous pairs of actors in a network (betweenness centrality), or as someone who has ties to actors who are highly central (eigenvector centrality). In some networks the same actor will score high on several measures. In others that will not be the case. In this chapter we will not consider all the various measures of centrality that researchers have developed but instead will focus on those that researchers tend to use the most. The chapter is divided into two main sections: the first focuses on those centrality measures that researchers use as estimates of power, while the second focuses on those that researchers use to estimate prestige.

6.1 Centrality and Power

Centrality in UCINET

Degree Centrality in UCINET

Network>Centrality

In UCINET all of the algorithms for estimating actor centrality are found in the *Network>Centrality* submenu. As we discussed in the previous chapter the most common (and oldest) measure of centrality is degree centrality, which in an undirected, dichotomous network[26] is simply a count of the number of ties that an individual actor has (i.e., the number of the neighbors). To begin with a simple example let us calculate the degree centrality of (the subset) of the business and marital ties between Renaissance Florentine families collected and recorded by John Padgett and Christopher Ansell (1993). Select the *Degree* centrality command found in the *Network>Centrality* submenu, which brings up a dialog box similar to Figure 6.1 below. Select the *Padgett.##h* file, accept UCINET's defaults (unless you want to change the name of the output file) and click OK.

Network>Centrality >Degree

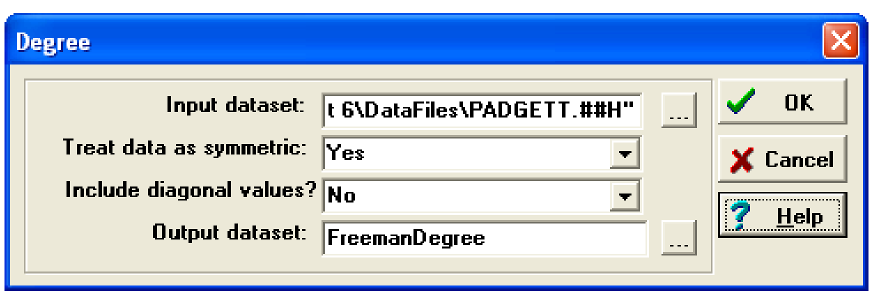
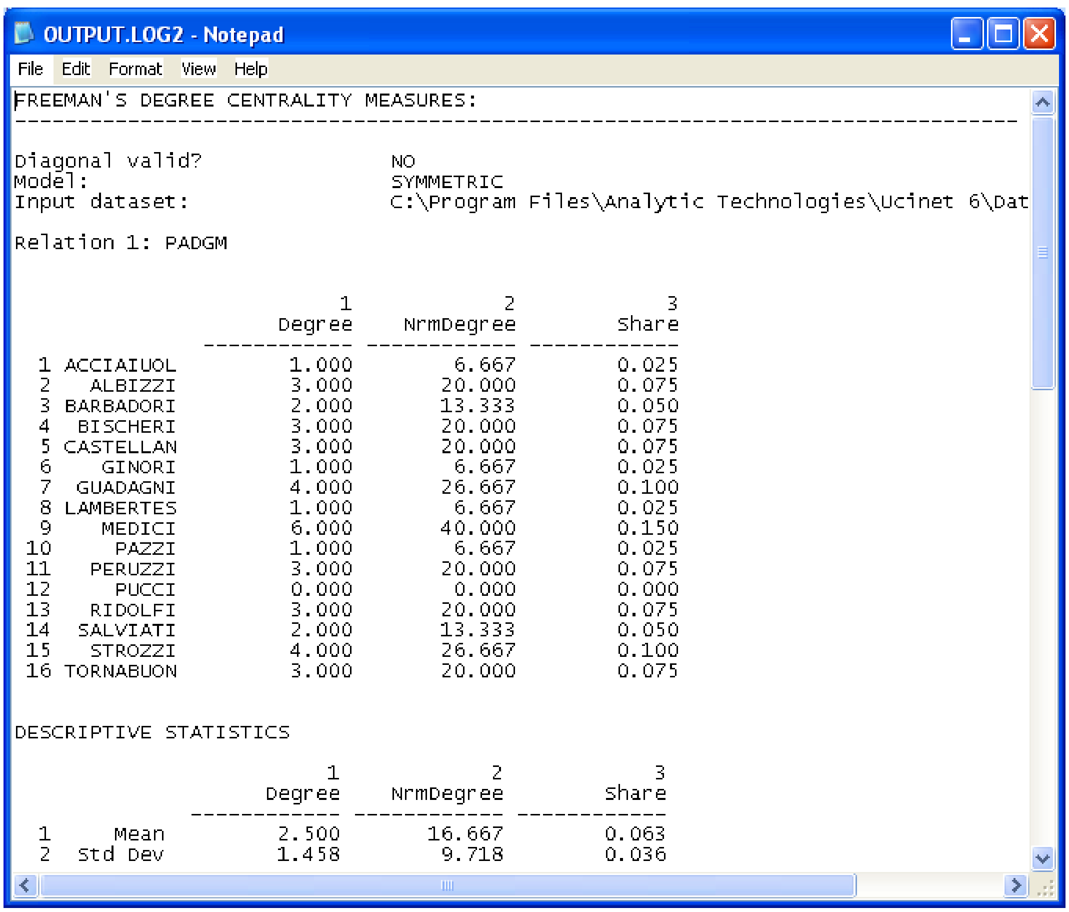


Figure 6.1: UCINET's Degree Centrality Dialog Box

[26] An undirected, dichotomous network is one that contains only edges (not arcs) and the presence or absence of a tie is indicated by either a "1" or "0."

This will call up a UCINET output log that should look similar to Figure 6.2 below. The first item listed is a table that itemizes the (1) degree centrality, (2) normalized degree centrality (expressed as a percentage for each actor), and the (3) share of each actor in the network. Normalized degree centrality is an actor's degree centrality score divided by the number of actors in the network minus one, while share is each actor's centrality measure divided by the sum of all of the actor centralities in the network (thus, they sum to one). The output log also includes a handful of descriptive statistics, such as the mean/average degree centrality (recall our use of average degree centrality in the previous chapter), standard deviation, variance, minimum value and maximum value. Following the descriptive statistics is the degree network centralization index expressed as a percentage (recall our use of network centralization in the previous chapter as well). Since the Padgett network data is a stacked matrix, if you scroll down the output log, you will discover that not only does the output include centrality measures for the marriage data but also for the business data. As you can see, the Medici is the most central family in terms of both marriage and business ties.


OUTPUT.LOG2 - Notepad

File Edit Format View Help

```
FREEMAN'S DEGREE CENTRALITY MEASURES:
--------------------------------------------------------------------------------

Diagonal valid?              NO
Model:                       SYMMETRIC
Input dataset:               C:\Program Files\Analytic Technologies\Ucinet 6\Dat

Relation 1: PADGM


                          1            2            3
                     Degree    NrmDegree        Share
               ------------ ------------ ------------
  1 ACCIAIUOL         1.000        6.667        0.025
  2   ALBIZZI         3.000       20.000        0.075
  3 BARBADORI         2.000       13.333        0.050
  4  BISCHERI         3.000       20.000        0.075
  5 CASTELLAN         3.000       20.000        0.075
  6    GINORI         1.000        6.667        0.025
  7  GUADAGNI         4.000       26.667        0.100
  8 LAMBERTES         1.000        6.667        0.025
  9    MEDICI         6.000       40.000        0.150
 10     PAZZI         1.000        6.667        0.025
 11   PERUZZI         3.000       20.000        0.075
 12     PUCCI         0.000        0.000        0.000
 13   RIDOLFI         3.000       20.000        0.075
 14  SALVIATI         2.000       13.333        0.050
 15   STROZZI         4.000       26.667        0.100
 16 TORNABUON         3.000       20.000        0.075


DESCRIPTIVE STATISTICS

                          1            2            3
                     Degree    NrmDegree        Share
               ------------ ------------ ------------
  1      Mean         2.500       16.667        0.063
  2   Std Dev         1.458        9.718        0.036
```

Figure 6.2: UCINET's Degree Centrality Output Log

Often, we will not be working with dichotomized data but with valued data where the value in each cell in the matrix represents the total number of ties between actors in the network (i.e., you can have more than one tie – kinship, religious, school – to another actor). In this case degree centrality equals the sum of the values of the ties. This means that UCINET's calculations of network centralization and normalized degree centrality will often yield scores greater than one, which is why with valued data you should focus only on the non-normalized values and ignore the degree centralization score. If you want to estimate the degree centralization of a valued network, you need to first dichotomize (i.e., binarize) the network (see below). Turning to the Noordin data, select the *Degree* centrality command found in the *Network>Centrality* submenu. At the dialog box, select the *Combined Noordin Network.##h* data file and accept UCINET's defaults (again, you may want to change the name of the output file) and click OK. The output log lists the degree centrality of each actor, which in this case, because the Noordin network is a valued network, equals the sum of the values of each actor's ties.

Network>Centrality >Degree

Now, let us assume that you only want to count the number of an actors' "neighbors," not the sum of the values of each actor's ties. In order to this, you need to first dichotomize the data where the cells with values greater than '0' are recoded to '1' and cell values of '0' are left alone. To do this, select the *Dichotomize* command found under the *Transform* menu, which brings up the following dialog box (see Figure 6.3). Load the *Combined Noordin Network.##h* data file, accept UCINET's defaults and click OK (note the name of the output dataset – default = *Combined Noordin NetworkGTO.##h*). Next, re-estimate degree centrality with the new dichotomized Noordin network. Now, if you choose, you can use the normalized degree centrality and network centralization scores.

Transform>Dichotomize

Network>Centrality >Degree

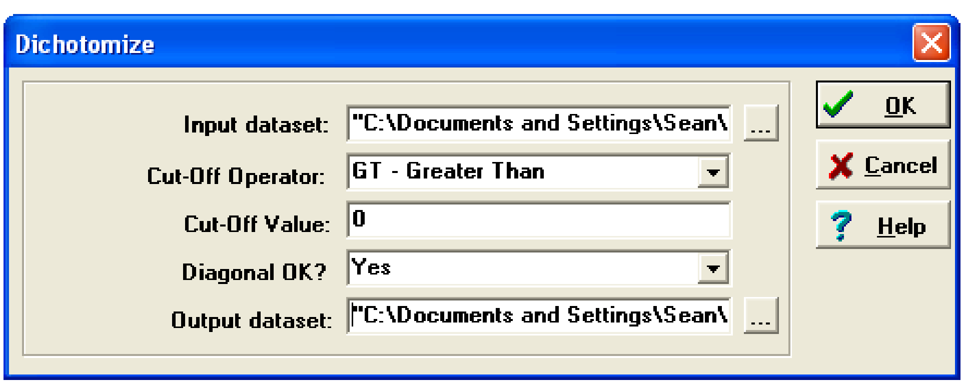


Figure 6.3: UCINET's Dichotomize Dialog box

Closeness Centrality in UCINET

Closeness centrality assumes that the closer an actor is to all other actors, the easier information may reach it, and as such enjoys higher centrality. An actor's closeness centrality is based on the total distance between one actor and all other actors, where larger distances yield lower closeness centrality scores. It is calculated by dividing the number of other actors in a network (i.e., N-1) by the sum of all distances between the actor and all other actors in the network. Closeness centrality cannot be calculated when a network includes isolated actors (i.e., actors that have no ties with other actors in the network) for the simple reason that no paths exist between isolated actors and the rest of the network. Thus, before calculating closeness centrality in a network, we need to first remove any isolates if, in fact, there are any. As it turns out the Padgett marriage and business data do contain isolates.[27] Thankfully, UCINET has made removing isolates relatively easy with its *Remove Isolates* command, which is found under the *Data>Remove* submenu. This brings up a dialog box (Figure 6.4) where you can select the Padgett data. Note what the new file will be called (or assign your own name) and click OK.

Data>Remove >Remove Isolates

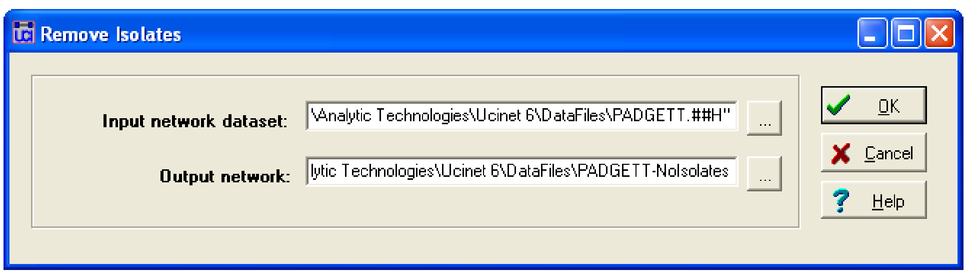


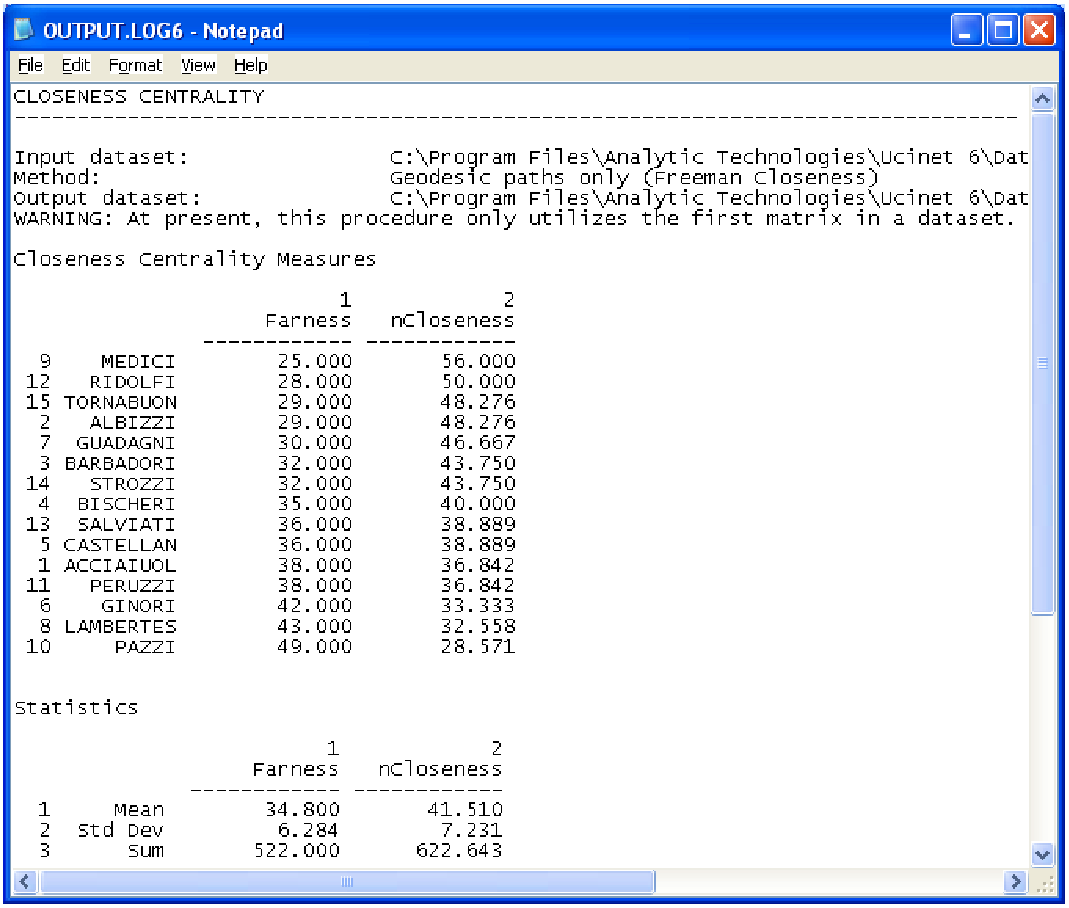
Figure 6.4: UCINET's Dichotomize Dialog box

Network>Centrality >Closeness

Next, calculate closeness centrality for the Padgett data using the *Network> Centrality>Closeness* command, being sure to use the newly-created (isolate-free) network. UCINET's output (Figure 6.5) is somewhat similar to its output when estimating degree centrality, but note that, unlike its degree centrality algorithm, UCINET *only calculates closeness centrality for the first matrix in the dataset*. Thus, if we want closeness scores for the Padgett business network, we would first need to extract the data and then estimate closeness centrality. Note that in the output lists/ranks the actors in terms of the closeness/farness: the actor that is closer, on average, to all other actors in the network is listed first while the actor that is farthest, on average, from all other actors is listed last. An actor's farness

[27] An easy way to tell whether a network contains isolates is when some of the actors have a degree centrality of zero.

score is the sum of the lengths of the geodesics to every other actor, while an actor's closeness score (nCloseness) is the total number of other actors (i.e., N-1; in this case 14) divided by the actor's farness score. As you can see the Medici family is closer (on average) to every other actor in the network. Like the output log for degree centrality, this one also includes descriptive statistics, such as the mean/average closeness centrality, standard deviation, variance, minimum value and maximum value. Following this the closeness network centralization index is listed.



```
CLOSENESS CENTRALITY
--------------------------------------------------------------------------------

Input dataset:                  C:\Program Files\Analytic Technologies\Ucinet 6\Dat
Method:                         Geodesic paths only (Freeman Closeness)
Output dataset:                 C:\Program Files\Analytic Technologies\Ucinet 6\Dat
WARNING: At present, this procedure only utilizes the first matrix in a dataset.

Closeness Centrality Measures

                               1            2
                         Farness   nCloseness
                    ------------ ------------
  9     MEDICI          25.000       56.000
 12    RIDOLFI          28.000       50.000
 15 TORNABUON           29.000       48.276
  2    ALBIZZI          29.000       48.276
  7   GUADAGNI          30.000       46.667
  3 BARBADORI           32.000       43.750
 14    STROZZI          32.000       43.750
  4   BISCHERI          35.000       40.000
 13   SALVIATI          36.000       38.889
  5 CASTELLAN           36.000       38.889
  1 ACCIAIUOL           38.000       36.842
 11    PERUZZI          38.000       36.842
  6     GINORI          42.000       33.333
  8 LAMBERTES           43.000       32.558
 10      PAZZI          49.000       28.571


Statistics

                              1            2
                        Farness   nCloseness
                   ------------ ------------
  1       Mean         34.800       41.510
  2    Std Dev          6.284        7.231
  3        Sum        522.000      622.643
```

Figure 6.5: UCINET's Closeness Centrality Output Log

Data>Remove >Remove Isolates

Network>Centrality >Closeness

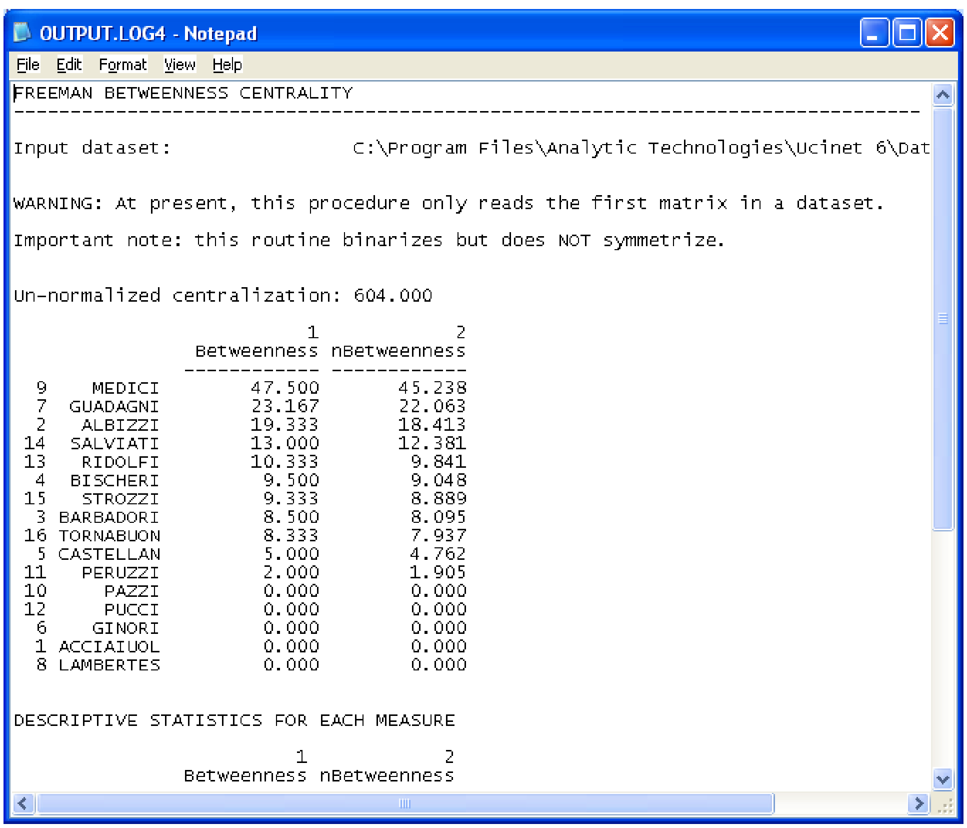
Now, replicate this procedure with the *Combined Noordin Network.##h* data file. First, we need to remove any isolates using UCINET's *Remove Isolates* command (noting the name of the newly-created data file). Next compute closeness centrality using the *Network>Centrality>Closeness* command. Which actors are the most central? Which ones are the least? How do these measures compare to the degree centrality scores computed earlier? Note that UCINET's output (not shown) indicates that before calculating closeness centrality, it dichotomizes the network. This means that if you were to repeat the process with

the *Combined Noordin NetworkGTO.##h* data, you should get the same result. You may want to try this in order to see this to be true.

Betweenness Centrality in UCINET

Network>Centrality >Freeman Betweenness >Node Betweenness

Betweenness centrality differs from degree centrality in that it assumes that an actor has power over any two other actors when it lies on the shortest path between them in a given network of relations. In order to calculate betweenness centrality for the *Padgett.##h* choose the *Freeman Betweenness>Node Betweenness* command found under the *Network>Centrality* submenu. As with closeness centrality, the output generated by UCINET (Figure 6.6) differs somewhat from that \when estimating degree centrality. For instance, like closeness centrality UCINET only calculates betweenness centrality for the first matrix in the dataset, which means that if we want betweenness scores for the Padgett business data, we would first have to extract the data and then estimate betweenness centrality. Still, some of the output is similar. UCINET provides both raw and normalized betweenness scores as well as a number of descriptive statistics, including network centralization based on betweenness.



```
OUTPUT.LOG4 - Notepad
File  Edit  Format  View  Help
FREEMAN BETWEENNESS CENTRALITY
--------------------------------------------------------------------------------

Input dataset:                C:\Program Files\Analytic Technologies\Ucinet 6\Dat

WARNING: At present, this procedure only reads the first matrix in a dataset.

Important note: this routine binarizes but does NOT symmetrize.

Un-normalized centralization: 604.000

                                1            2
                      Betweenness nBetweenness
                     ------------ ------------
  9     MEDICI         47.500       45.238
  7   GUADAGNI         23.167       22.063
  2    ALBIZZI         19.333       18.413
 14   SALVIATI         13.000       12.381
 13    RIDOLFI         10.333        9.841
  4   BISCHERI          9.500        9.048
 15    STROZZI          9.333        8.889
  3  BARBADORI          8.500        8.095
 16  TORNABUON          8.333        7.937
  5  CASTELLAN          5.000        4.762
 11    PERUZZI          2.000        1.905
 10      PAZZI          0.000        0.000
 12      PUCCI          0.000        0.000
  6     GINORI          0.000        0.000
  1  ACCIAIUOL          0.000        0.000
  8  LAMBERTES          0.000        0.000


DESCRIPTIVE STATISTICS FOR EACH MEASURE

                              1            2
                    Betweenness nBetweenness
```

Figure 6.6: UCINET's Betweenness Centrality Output Log

Which actor is the most central actor in terms in betweenness centrality? The second? Third? Fourth? Fifth? Now try the same routine with the Combined Noordin NetworkGTO.##h network. Are the scores any different? You may have already noticed in the output that like UCINET's closeness centrality routine, UCINET binarizes (i.e., dichotomizes) the network before estimating betweenness centrality, which means that the scores using the *Combined Noordin NetworkGTO.##h* data should be the same as those for the *Combined Noordin Network.##h*.

Eigenvector Centrality in UCINET

Network>Centrality >Eigenvector

Eigenvector centrality assumes that ties to highly central actors are more important than are ties to peripheral actors. Thus, it weights an actor's centrality by the centrality scores of its neighbors (i.e., the actors to which it has ties). To compute eigenvector centrality in UCINET, select the *Eigenvector* command under the *Network>Centrality* submenu. UCINET's output first lists a series of eigenvalues before listing the eigenvector centrality scores for each actor. Like the output for degree and closeness centrality, it provides a number of descriptive statistics, including network centralization based on closeness.

Summary: Multiple Centrality Measures in UCINET

Network>Centrality >Multiple Measures

UCINET includes a handy *Multiple Measures* command under the *Network>Centrality* submenu, which calculates degree, closeness, betweenness and eigenvector centrality. However, while it estimates a number of descriptive statistics, it does not calculate centralization scores for the various types of centrality. When you use this command (regardless of the type of data), the output log will indicate that before estimating any of the measures, UCINET automatically dichotomizes the data. Moreover, it only reports normalized (rather than raw) centrality scores (the output log does not tell you this, however). Thus, the scores you get from this routine may not always agree with the scores estimated using the other commands, which report both raw and normalized scores and do not always dichotomize the data. That is why you need to be careful when using this command. While it is very convenient, you need to keep in mind the assumptions UCINET uses when calculating the measures!

As mentioned at the outset of this chapter, these are not the only centrality measures that UCINET calculates. It also calculates alpha (aka, power centrality), influence, hubs and authorities, reach, information, proximal betweenness, flow betweenness, fragmentation and contribution centrality measures. If you want to

learn more about the assumptions lying behind these measures, you can either consult Hanneman and Riddle (2005) or consult UCINET's help function.

Centrality in NetDraw

Centrality measures of individual actors are essentially attributes of actors, which means that we can visualize them as attributes in network mapping programs such as *Pajek* and *NetDraw*. In this section we will first examine how to use centrality scores calculated in UCINET to modify our visualization of networks in NetDraw. Then, we will see that NetDraw also includes a feature for calculating centrality scores that can then be utilized in our visualizations.

Visualizing UCINET Centrality Scores in NetDraw

[NetDraw] File>UCINET dataset >Network

File>UCINET dataset >Attribute data

Transform >Node Attribute Editor

Properties>Nodes >Symbols>Size >Attribute-based

In *NetDraw* open the *Combined Noordin Network.##h* data file and energize it choosing one of NetDraw's layout algorithms. Next open the centrality attribute files associated with this data (i.e., FreemanDegree, Freeman Betweenness, Closeness, EigenvectorCentrality). Note, however, that the closeness data will not necessarily match up with your network data because before we calculated it in UCINET, we eliminated isolates. Thus, after opening it you will need to examine the data (using the *Node Attribute Editor* command found under the *Transform* menu) to see if it matched up correctly with the actors (e.g., Nasir Abas should have a closeness centrality score of '0' because that is the one isolate in the network). Then, using the *Properties>Nodes>Size>Attribute-based* command (or *Properties>Nodes>Symbol>Size>Attribute-based* command, depending on what version of UCINET), select various centrality measures (in the Size of Nodes dialog box – See Figure 6.7, next page) to vary the size of the nodes. What measure of centrality do you find to be the most illuminating? Why?

Visualizing UCINET Centrality Scores in NetDraw

Analysis >Centrality measures

Transform >Node Attribute Editor

NetDraw also calculates centrality measures with its *Analysis>Centrality measures* command. This brings up a dialog box where you can choose which centrality measures you want NetDraw to calculate. NetDraw's default is to calculate all of the measures (degree, closeness, harmonic closeness, betweenness, eigenvector and 2-local eigenvector). If you first click OK and then open the *Node Attribute Editor*, you will see that NetDraw places the calculated scores in the attribute editor. You may want to compare the calculations made by UCINET and NetDraw. In theory all should be the same; however, that is not always the case. If the scores do not agree, then it is advisable to use those calculated in UCINET.

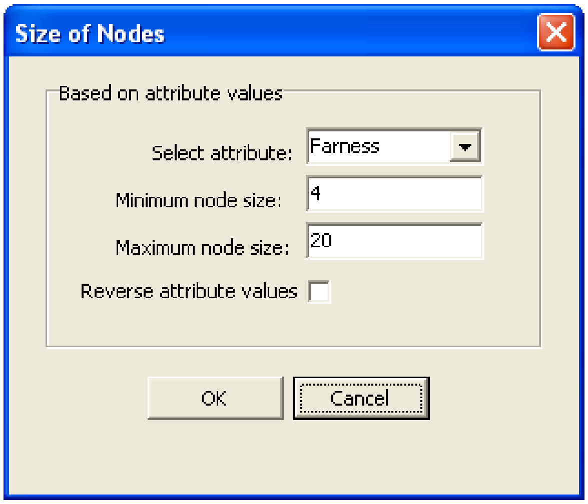


Figure 6.7: NetDraw's Size of Node dialog box

Another thing to keep in mind when estimating centrality in NetDraw is that if you have imported a stacked matrix (i.e., where multiple types of ties are listed in the relations box found on the right side of NetDraw), NetDraw will only calculate centrality on the relations that are checked. For example, if in NetDraw you open Krackhardt's High Tech data and only select the Advice network, NetDraw will only calculate centrality for the Advice network. If you select the Advice and Friendship network, it will calculate centrality on the combined Advice and Friendship network, and so on. Thus, you must always be aware of what you are doing when issuing commands whether you are using NetDraw, UCINET or Pajek.

Centrality in Pajek

Estimating centrality in Pajek is relatively straightforward although to date it has only implemented three centrality algorithms: degree, closeness and betweenness. It is possible to visualize other measures of centrality in Pajek, but you need to estimate the scores in another program such as UCINET and then import the scores into Pajek as either a partition or vector.

Degree Centrality in Pajek

File>Pajek Project File >Read

Net>Transform >Arcs→Edges>All

Net>Partitions >Degree >Input, Output, All

Read the *Combined Noordin Network* project file into Pajek. Pajek treats degree centrality as a discrete, rather than as a continuous, attribute of an actor (i.e., it is always an integer), so *Pajek* stores it as a partition. To calculate degree centrality in *Pajek*, choose either the *Input*, *Output* or *All* option found under the *Net>Partitions> Degree* submenu. *Input* counts all incoming lines (indegree), *Output* counts all outgoing lines (outdegree), and *All* counts both. Note that an edge, which has no direction, is considered to be incoming as well as outgoing, so each edge is counted once by all three commands. Thus, in an undirected network it makes no difference whether you select *Input*, *Output*, or *All*. However, because when an undirected network is exported from UCINET and read into Pajek, edges are often transformed into arcs, you will probably want to symmetrize your data (i.e., make it symmetric) *before* calculating degree centrality, using the *Net> Transform>Arcs→Edges>All* command, unless, of course, you are working with a directed network (see discussion of using directed networks for calculating measures of prestige below). Pajek will ask whether you want to create a new network (select yes) and whether you want to remove multiple lines and loops (select options 1, 2, 3 or 4). After doing this, estimate degree centrality. Pajek will create a new partition, which assigns each to a particular class (the number of which equals its degree centrality); it also creates a vector of normalized degree centrality scores.

Draw >Draw-Partition-Vector

Options>Value of Lines >Similarities

Options>Size>of Vertices

Layers >Type of Layout>2D

Layers>In y direction

With the “All Degree partition of N1 (79)” showing in the partition dropdown list and the “Normalized All Degree partition of N1 (79)” highlighted in the vector dropdown list, view the network with node size adjusted for degree centrality by selecting the *Draw>Draw-Partition-Vector* command and then using one of the 2D layout algorithms. Be sure that the *Options>Value of Lines>Similarities* command is checked. Also, if the node sizes do not seem to vary in size, then you may need to adjust the *Size>of Vertices* option under the *Options* menu. Select ‘0’ to tell Pajek to automatically adjust the size of the nodes. Next, after telling Pajek that this is a 2D layout with the *Layers>Type of Layout>2D* command found in the Draw screen, instruct Pajek to layer the drawing in the “y-direction” using the *Layers>In y direction* command. This should place Noordin at the bottom of the drawing (i.e., the person with the highest degree centrality) and Nasir Abas (i.e., the person with the lowest degree centrality) at the top. If you hold down the “X” key, you can rotate the drawing so that Noordin is at the top. Now your drawing is layered in terms of degree centrality with everyone having the same degree centrality located at the same horizontal level (see Figure 6.8 below).

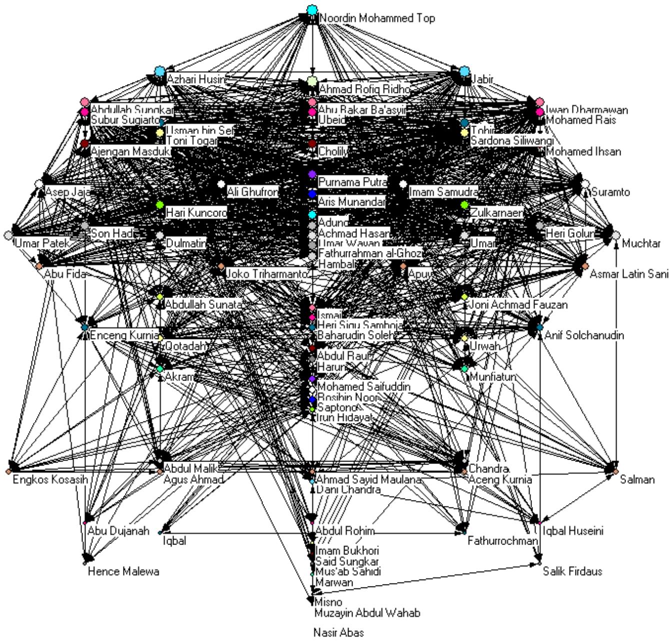


Figure 6.8: Pajek Drawing of Noordin's Network, Layered by Degree Centrality

Closeness Centrality in Pajek

[Main Screen] Net>Vector>Centrality >Closeness >Input, Output, All

In Pajek, the computation of closeness and betweenness centrality is similar to that of degree, except that Pajek treats both as vectors rather than partitions because they are continuous rather than discrete measures. Consequently, the centrality command for both are located in *Net>Vector>Centrality* submenu. The command to compute closeness centrality for all vertices in the network includes *Input*, *Output* and *All* options which, with an undirected network, yield the same results. If the network is disconnected (i.e., where there are isolates), Pajek assigns isolated actors a score of '0'. To see the scores of individual actors, select the *File>Vector>Edit* command (or select the "Edit" radio button). This will call up an editing box that allows you to not only scroll down through the various scores assigned to each actor but also edit the scores if you so choose. You may want to compare these scores to those calculated in UCINET and NetDraw. Also, open Pajek's report window (*File>Show Report Window*) and see what Pajek says about the calculation of network centralization. Finally, use Pajek's *Info>Vector* command (accept Pajek's defaults in the dialog boxes that appear) to get basic information about the closeness centrality scores.

File>Vector>Edit

File>Show Report Window

Info>Vector

We can visualize the closeness scores in much the same way as we did with the degree centrality scores. With the *Combined Noordin Network* showing in the network dropdown list and the "Closeness Centrality in N1" vector showing in the vector dropdown list, view the network with node size adjusted for closeness

Draw>Draw-Vector

centrality by selecting the *Draw>Draw-Vector* command and using one of the 2D layout algorithms. Note that there appears to be very little variation in node size, suggesting that closeness centrality measures do not appear to vary too much

Betweenness Centrality in Pajek

Net>Vector>Centrality >Betweenness

Draw>Draw-Vector

Betweenness centrality is calculated and visualized in Pajek in essentially the same way as estimating closeness centrality, using the *Net>Vector>Centrality >Betweenness* command. Note, however, that Pajek does not include In, Out and All options for betweenness. This is because betweenness centrality does not take into consideration the direction of ties, only the path length between actors. When you issue this command, Pajek creates a vector that you can examine using Pajek's *File>Vector>Edit*, *File>Show Report Window* and *Info>Vector* commands or view with the Noordin network using the *Draw>Draw-Vector* command.

6.2 Centrality, Power and Prestige

Social network researchers generally assume that a tie pointing at someone is a measure of prestige, which is why they typically use *indegree centrality* as a measure of prestige. For example, a member of a dark network to whom people go to for advice (e.g., a mentor) could be seen as enjoying higher levels of prestige in the network than those who only seek advice from others. As we will see, there are variations on this approach, but indegree centrality is typically the starting point.

To illustrate how to estimate prestige, we will use data collected by David Krackhardt from the managers of a high-tech company that manufactured high-tech equipment on the West Coast. According to the description of the dataset in UCINET 6, at the time of Krackhardt's study, the firm had been in existence for ten years, produced high-tech machinery for other companies and employed approximately 100 people of which 21 were managers (see also, Krackhardt 1987, 1992). The managers are the actors in the dataset. Krackhardt gave each manager a roster of the names of the other managers and asked to check the other managers to whom they would go for advice at work ("Advice"), and with whom they were friends ("Friends").[28] He also collected cognitive social structure data from the 21 managers to assess the effects of a recent management intervention program. The relation queried was "Who does X go to for advice and help with work?" (Krackad) and "Who is a friend of X?" (Krackfr). Thus, not only did each person

[28] Krackhardt also collected data on "who reports to whom" for all 21 managers ("Reports to").

indicate his or her own advice and friendship relationships but also the relations he or she perceived among all other managers, generating a full 21 by 21 matrix of adjacency ratings from each person in the group.

Estimating Prestige in UCINET

Data>Extract >Extract submatrix

Because Krackhardt's data is recorded in three stacked 21x21 matrices, the first thing we need to do is extract the matrices from one another. In UCINET select the *Extract>Extract submatrix* command from the *Data* menu.[29] This will bring up the Extract dialog box (see Figure 6.9). Click on the "…" button to the right of the "Input dataset" drop list and select the Krack-High-Tec dataset.

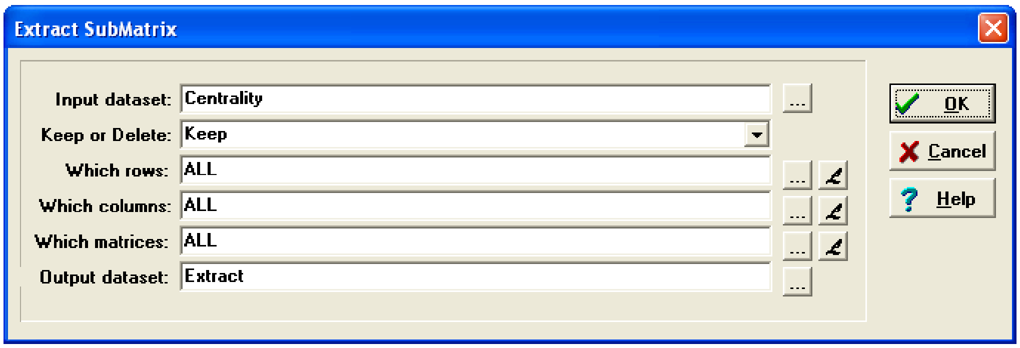


Figure 6.9: UCINET's Extract Submatrix Dialog Box

Next, click on the ***L*** (Labels) button to the right of the "Which matrices" drop list. This will bring up the "Select labels" dialog box. Highlight "Advice" and click OK. You have now selected Krackhardt's Advice matrix as the dataset that you will extract. Before doing so, you will want to give a name to the extracted network – I recommend something like "Krack-Advice" – type the name you choose in the "Output dataset" drop list. The "Extract" dialog box should look something like the following figure. Click on the OK button when you are ready. Repeat the process and extract Krackhardt's Friendship Network. Be sure to give this network an easily recognizable name as well.

Network>Centrality >Degree

To calculate indegree centrality in UCINET, first select the *Degree* option found under the *Network>Centrality* submenu. This will bring up UCINET's Degree Centrality dialog box. Make sure that the "Krack-Advice" network is highlighted in the "Input dataset" option box. Next, select the "No" option in the "Treat data as symmetric" option box, which tells UCINET to calculate both the indegree and outdegree centrality scores of each vertex in a network. (When you select "Yes," it calculates and then selects the higher of the indegree or outdegree

[29] We could also use UCINET's *Data>Unpack* command.

centrality of each vertex). Looking at the Krackhardt Advice network, which manager enjoys the highest level of prestige? Repeat this step using Krackhardt's Friendship network. In terms of the friendship network, which vertex enjoys the highest level of prestige? Export the Friendship network in Pajek format (e.g., Krack-Friend.net) using the *Data>Export>Pajek>Network* command, and then open Pajek. Remember to open Pajek through Windows' Start menu rather than using the version of Pajek that comes with UCINET.

Data>Export >Pajek>Network

Estimating Prestige in Pajek

File>Network>Read

Net>Partitions >Degree>Input

Info>Partition

Open the Friendship network in Pajek. Select the *Net>Partitions>Degree> Input* command, which creates a new partition that you can display with the *Info>Partition* command (accept the default's provided by Pajek). Be sure that the new partition is showing in the partition drop list. This should create a report that looks something like the following figure (Figure 6.10). The cluster number tells you the indegree centrality of that cluster/class. Thus, there are 2 vertices (9.5238% of total) with an indegree centrality of 1, which is the lowest in the network, while only 1 vertex has an indegree centrality of 10, which is the highest in the network. How many vertices have an indegree centrality of 5?

```
----------------------------------------------------------------------
l. Input  Degree partition of N1 (21)
----------------------------------------------------------------------
)imension: 21
[he lowest value: 1
[he highest value: 10

?requency distribution of cluster numbers:

 Cluster      Freq     Freq%   CumFreq  CumFreq% Representative
 ---------------------------------------------------------------
       1         2    9.5238         2    9.5238 10
       2         1    4.7619         3   14.2857 6
       3         2    9.5238         5   23.8095 7
       4         3   14.2857         8   38.0952 15
       5         6   28.5714        14   66.6667 3
       6         4   19.0476        18   85.7143 5
       8         2    9.5238        20   95.2381 1
      10         1    4.7619        21  100.0000 2
 ---------------------------------------------------------------
    Sum         21  100.0000
```

Figure 6.10: Pajek's Report Window

File>Partition >Edit

Close the report menu and select the *File>Partition>Edit* command. This will bring up a window that displays the indegree centrality measures of each vertex in the network.

Alternative Prestige Measures Available in Pajek

As de Nooy, Mrvar and Batagelj (2005) note, indegree centrality is a somewhat restricted measure of prestige because it only considers direct choices. It does not matter whether choices are received from people who are not chosen themselves or from popular people. In other words, the overall structure of the network is disregarded. As a result the developers of Pajek have included routines for two additional measures that can be used to estimate actor prestige: input domain and proximity prestige.

Net>Partitions >Domain>Input

Input domain is a measure of prestige that counts all people by whom someone is nominated whether directly or indirectly. The larger a person's input domain, the higher his or her prestige. To calculate input domain use Pajek's *Net>Partitions>Domain>Input* command. The command *Input* restricts the analysis to incoming arcs only. A dialog box allows you to specify a maximum distance for the input domain. For now, accept Pajek's default (0 – No limit). This command produces one partition and two vectors. The partition specifies the number of vertices within the input domain of each vertex. The vector labeled 'Normalized Size of input domain' lists the size of input domains as a proportion of all vertices (minus the vertex itself), and the second vector gives the average distance to a vertex from all vertices in its input domain.[30]

Info>Partition

File>Partition>Edit

As you did before, use the *Info>Partition* command to examine the partition (Again, be sure that the new partition is showing in the partition drop list). Close the report menu and select the *File>Partition>Edit* command, which as we saw above, brings up a window that displays the indegree centrality measures of each vertex in the network. Which vertices have the highest input domain?

Net>Partitions >Domain>Input

The input domain of a vertex is a far from perfect measure of prestige. In a well-connected network, the input domain of a vertex often contains all or almost all other vertices, so it does not distinguish very well between vertices. One solution is to limit the input domain to direct neighbors or to neighbors at a maximum distance of two on the assumption that nominations by close neighbors are more important than nominations by distant neighbors. Repeat the above commands except this time choose an input domain of 2 (i.e., a friend of a friend). Is there more variation in input domain than before? Now, which vertices have the highest input domain?

[30] It is impossible to compute average distance in the case of a vertex with an empty input domain, that is, a vertex which is not chosen at all. In this case, average distance is set to 999998, which represents infinity.

Needless to say, the choice of a maximum distance from neighbors within a restricted input domain is quite arbitrary. The concept of proximity prestige overcomes this by considering all vertices within the input domain of a vertex but attaching more importance to a nomination if it is expressed by a closer neighbor. That is, a nomination by a close neighbor contributes more to the proximity prestige of an actor than does a nomination by a distant neighbor. At the same time many 'distant nominations' may contribute as much as one 'close nomination'. In short, proximity prestige weights each choice by its path distance to the vertex.

To calculate proximity prestige, we need to divide the input domain size by the average distance. To do this first be sure that the vector with the normalized size of the input domain – "Normalized Size of input domain in N1 (21)" – is highlighted in the first vector drop list and the vector with average distances – "Average distance from input domain in N1 (21)" – is highlighted in the second vector drop list. Then, click on the command *Divide First by Second* in the *Vectors* menu. This will create a new vector containing the proximity prestige scores of all vertices. Inspect them with the command *Info>Vector* or browse with *File>Vector>Edit*. Proximity prestige scores must range from zero to one. If they do not, you probably specified the wrong vectors in the *Vectors* menu. Which vertex has the highest proximity prestige score? Is this the same vertex that has the highest indegree centrality score?

Vectors >Divide First by Second

Info>Vector

File>Vector>Edit

6.3 Summary

In this chapter we have seen that centrality is one of the oldest concepts in social network analysis. It has roots in the sociometric stars of Jacob Moreno (1953), the experiments of Alex Bavelas (1950) and Richard Emerson and Karen Cook (Cook and Emerson 1978; Cook et al. 1983; Emerson 1962, 1972a, 1972b, 1976) and the mathematical insights of Linton Freeman and Phil Bonacich (Bonacich 1987; Freeman 1979). We have also seen that a central actor can be conceptualized in a variety of ways; however, in this chapter we did not consider all the various measures of centrality that researchers have developed but instead focused on those that researchers tend to use the most. We also explored ways that researchers use measures of centrality to estimate the prestige of actors within a network.

CHAPTER 7

COHESION AND CLUSTERING

A major focus of social network analysis is to identify dense clusters of actors "among whom there are relatively strong, direct, intense, and/or positive ties" (Wasserman and Faust 1994:249). Social network analysts often refer to these clusters of actors as cohesive subgroups and generally assume that "social interaction is the basis for solidarity, shared norms, identity, and collective behavior, so people who interact intensively are likely to consider themselves a social group" (de Nooy, Mrvar and Batagelj 2005: 61). Social network analysts use several approaches for identifying cohesive subgroups. One way is to cluster actors based on attributes (e.g., race, gender, etc.). Another is to focus on the pattern of ties (i.e., relations) among actors. In an ideal world there would be one method that we could use to identify cohesive subgroups. However, we do not live in an ideal world, so it is not surprising that social network analysts have developed a variety of methods for identifying clusters of actors.

> Once analysts began to try to formalize the idea of the clique and to devise mathematical measures of the number and cohesion of cliques, it was… recognized that there were a number of different ways of operationalizing the apparently simple idea of the 'clique': for example, cliques could be seen as groups of mutually connected individuals or as pockets of high density. Thus, a number of different theoretical models of subgroups emerged, variously described as 'cliques', 'clusters', 'components', 'cores' and 'circles'. Apart from beginning with the letter 'c', these concepts have very little in common with one another (Scott 2000:100).

This chapter explores some (but not all) of the various approaches for using patterns of ties for identifying cohesive subgroups within social networks. See Hanneman and Riddle (2005) for an exhaustive review of the various cohesive subgroup algorithms available in UCINET.

7.1 Components

Probably the simplest form of subgroup are components, which are subnetworks where members have ties to one another but do not have ties with members of other subnetworks (Hanneman and Riddle 2005). In directed networks, you can identify two types of components: strong and weak. Strong components take into consideration

the direction of the ties whereas weak components do not. In a strong component each pair of vertices is connected by a (directed) path and no other vertex can be added without destroying its connectedness. By contrast, in a weak component each pair of vertices is connected by an undirected path (i.e., a semi-path) and no other vertex can be added without destroying its connectedness. Take, for example, the network in Figure 7.1 below. The network contains three strong components – (1) actor 1, (2) actor 2 and (3) actors 3, 4 & 5 – and two weak components – (1) actor 2 and (2) actors 1, 3, 4 & 5. Actors 1, 3, 4 & 5 do not constitute a strong component because you cannot "walk" from actor 3, 4 or 5 to actor 1 following the direction of the arrows. However, they do constitute a weak component because you can "walk" from actor 3, 4 or 5 to actor 1 if you ignore the direction of the arrows. As this example illustrates, strong connectedness is more restrictive than weak connectedness. Thus, while each strongly connected network is also weakly connected, each weakly connected network is not necessarily strongly connected (de Nooy, Mrvar and Batagelj 2005:67). Figure 7.1 also illustrates that weak components are (by definition) isolated from one another, which suggests that attempting to identify components in a well-connected and undirected network may not prove a good use of one's time.

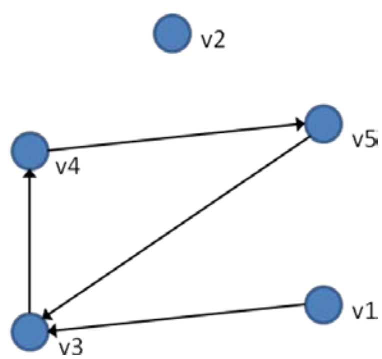


Figure 7.1: Simple Unconnected Directed Network

Identifying Components in UCINET

We will initially examine a dataset collected in 1948 by sociologists who conducted a large study in the Turrialba region of Costa Rica (Loomis et al. 1953).[31] Among other things, they recorded the network of visiting relations between families living in farms in a neighborhood called Attiro. The network of visiting relations is a simple directed graph where each arc represents "frequent

[31] De Nooy, Mrvar and Batagelj (2005) use this dataset in Chapters 3 and 9 of their book, *Exploratory Social Network Analysis with Pajek*. We analyze it here because it lends itself quite well to exploring cohesive subgroups.

visits" from one family to another. Line values do not represent the exact number of visits but rather indicate the type of visiting relation: ordinary ("1"), visits among kin ("2"), and visits among ritual kin (i.e., between god-parent and god-child – "3"), but we will ignore them in this chapter. Loops do not occur (i.e., the diagonal = 0) because people do not visit themselves.

[UCINET] Network>Regions> Components >Simple graphs

To detect components in UCINET we use the *Network>Regions> Components>Simple graphs* command, which brings up the dialog box below (Figure 7.2). Since the Attiro network is directed, under the "Kind of components" drop down box, I selected the "Strong" option and changed the name of two of the output files; otherwise I accepted UCINET's defaults, including the option to only save components of size three or greater for the simple fact that components smaller than this are relatively uninteresting. Note that the dialog box indicates that UCINET produces four different output files, [32] two of which will concern us here: (1) the partition matrix and (2) the main component vector. The "partition matrix" indicates the component of which each actor is a part (if at all – remember we accepted UCINET's default of setting the minimum component size to 3). We will use this file for visualization purposes in NetDraw. The main component vector is another partition that we can use to identify which actors belong to the largest (i.e., main) component. We will use this for visualization purposes as well.

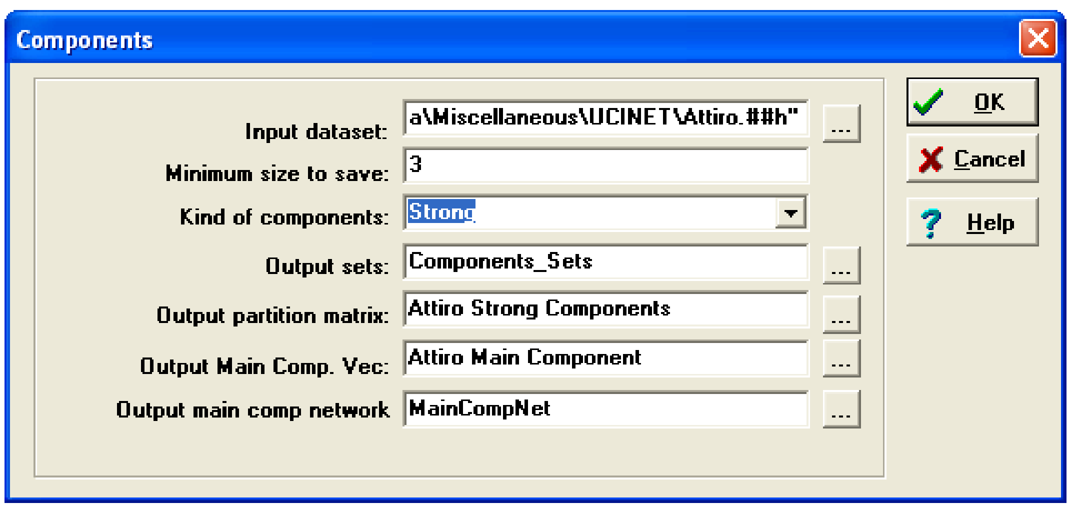
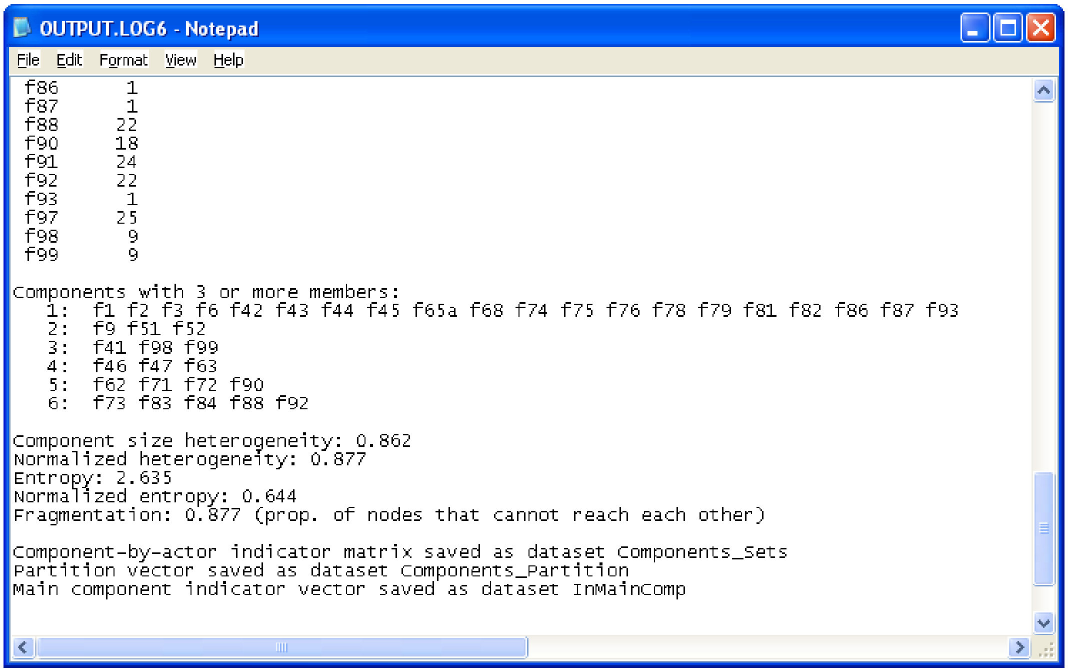


Figure 7.2: UCINET Components Dialog Box

Clicking on the OK button yields an output log (not shown) that first lists the various components and their sizes (including components with sizes less than the minimum size indicated in the dialog box). It then lists the components to which

each of the actors (i.e., nodes) belongs (again including components with sizes less than the indicated minimum size). Finally, it lists the components that are the minimum size (in this case three) or larger. This portion of the output log is shown in Figure 7.3 below. Note that the output indicates that there are six strong components of size three or greater, the first one of which includes 20 families.



OUTPUT.LOG6 - Notepad

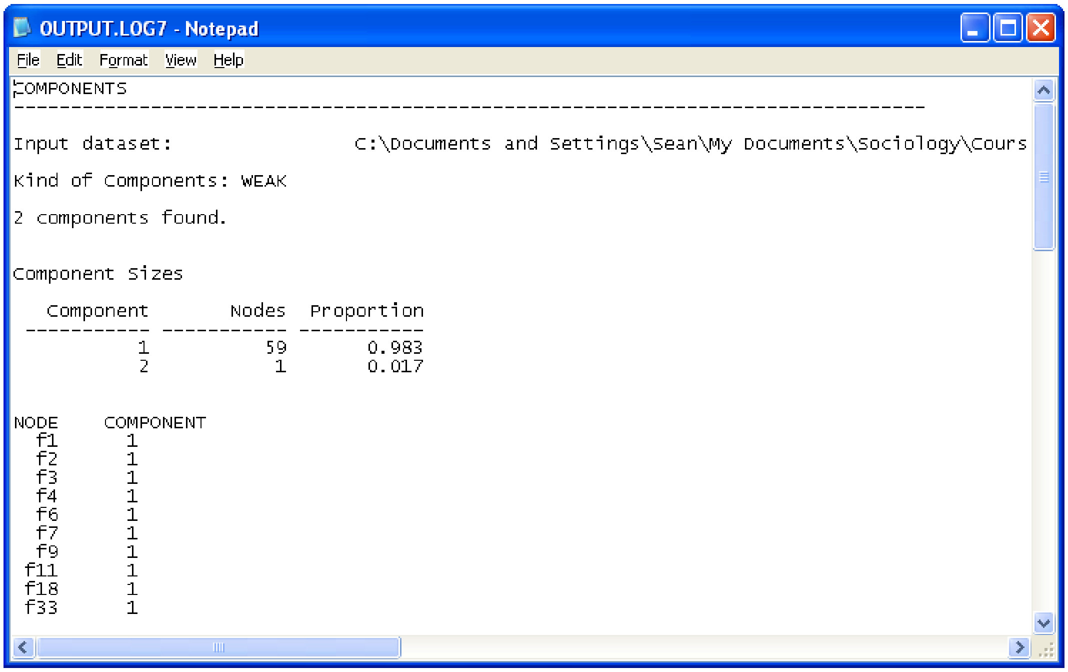
File Edit Format View Help

```
 f86      1
 f87      1
 f88     22
 f90     18
 f91     24
 f92     22
 f93      1
 f97     25
 f98      9
 f99      9

Components with 3 or more members:
   1:  f1 f2 f3 f6 f42 f43 f44 f45 f65a f68 f74 f75 f76 f78 f79 f81 f82 f86 f87 f93
   2:  f9 f51 f52
   3:  f41 f98 f99
   4:  f46 f47 f63
   5:  f62 f71 f72 f90
   6:  f73 f83 f84 f88 f92

Component size heterogeneity: 0.862
Normalized heterogeneity: 0.877
Entropy: 2.635
Normalized entropy: 0.644
Fragmentation: 0.877 (prop. of nodes that cannot reach each other)

Component-by-actor indicator matrix saved as dataset Components_Sets
Partition vector saved as dataset Components_Partition
Main component indicator vector saved as dataset InMainComp
```

Figure 7.3: UCINET (Strong) Components Output Log

What happens if we treat the network as an undirected network by indicating that we want weak components rather than strong components? Figure 7.4 (next page) displays the answer. When the direction of the network is ignored, UCINET only detects two components – one with 59 families and one with 1 family – the family that has no connections to any of the other families (at least in terms of the visiting network). As you can see the components routine is much better at identifying subgroups when it works with directed data rather than undirected data. You can see this for yourself with the Noordin data. Take the "Combined Noordin Network" file and issue UCINET's *Network>Regions> Components>Simple graphs* command. You will discover that there is only one component of size three or greater, regardless of whether you use the "Strong" or "Weak" option. This illustrates the value of collecting directed data whenever possible (e.g., email and phone communication, the flow of financial and other types of resources).

[32] Interestingly, the output log only indicates that the output only includes three files. It does not appear that UCINET currently outputs the fourth file indicated by the dialog box.



```
OUTPUT.LOG7 - Notepad
File  Edit  Format  View  Help
COMPONENTS
------------------------------------------------------------------------------

Input dataset:              C:\Documents and Settings\Sean\My Documents\Sociology\Cours

Kind of Components: WEAK

2 components found.


Component Sizes

   Component       Nodes  Proportion
 ----------- ----------- -----------
           1          59       0.983
           2           1       0.017


NODE    COMPONENT
  f1      1
  f2      1
  f3      1
  f4      1
  f6      1
  f7      1
  f9      1
 f11      1
 f18      1
 f33      1
```

Figure 7.4: UCINET (Weak) Components Output Log

Visualizing Components in NetDraw

[NetDraw] File>Open>Ucinet dataset>Network

File>Open>Ucinet dataset>Attribute data

Properties>Nodes>Symbols >Color>Attribute-based

Now let us see how NetDraw can help us visualize components. In NetDraw first open the "Attiro" network file using NetDraw's *File>Open>Ucinet dataset> Network* command. Then open both the Attiro Strong Components and Attiro Main Component files using NetDraw's *File> Open>Ucinet dataset>Attribute data* command. Finally, use the *Properties>Nodes>Symbols>Color>Attribute-based* command so that NetDraw assigns each actor a different color based on the component to which it belongs (by choosing the "Strong Components" attribute. This should yield a picture similar to Figure 7.5 (next page). Note that the strong components partition created by UCINET includes more than the six components of size three or greater. It includes all of the components identified by UCINET. Thus, it is hard to visually distinguish the six largest components in the network. We can, however, identify the largest (i.e., the main) component quite easily by assigning colors to actors using the "Main Component" attribute (after issuing the *Properties>Nodes>Symbols>Color>Attribute-based* command). This will yield a picture similar to Figure 7.6 (next page) where the red-colored actors belong to the main component.

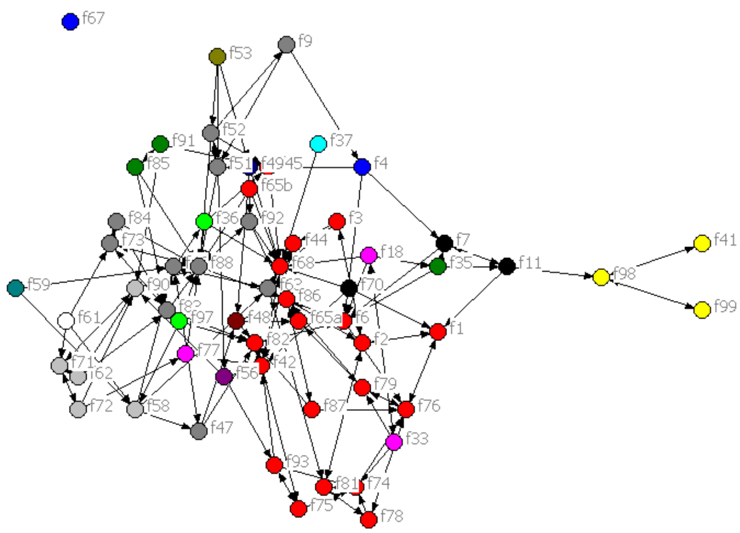


Figure 7.5: NetDraw Visualization of the Attiro Network's Strong Components

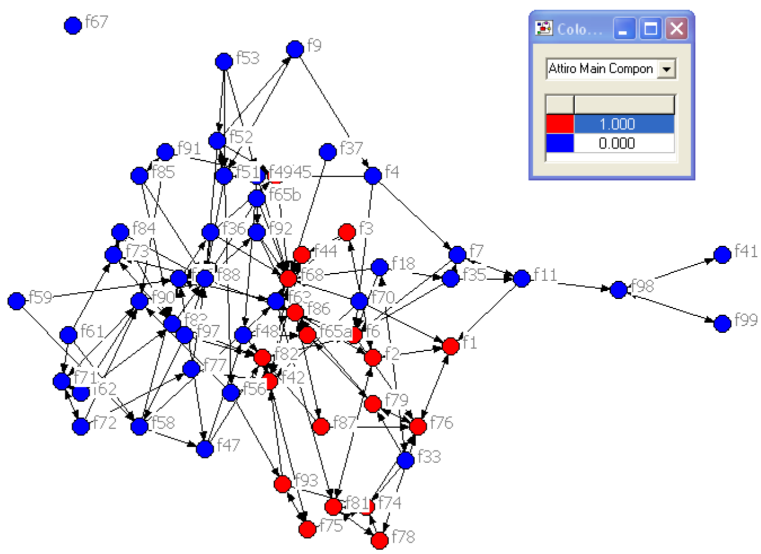


Figure 7.6: NetDraw Visualization of the Attiro Network's Main Component

Finally, NetDraw can identify components with its *Analysis>Components* command. However, be advised that it only identifies "weak," not "strong," components as Figure 7.7 (next page) indicates. Note that all of the actors are colored red except the one isolate in the network in the upper left corner of the network map. This is not an issue if you are analyzing an undirected network; however, if you are working with a directed network and you analyze the network in NetDraw (and not in UCINET or Pajek), then you may fail to indentify a network's cohesive subgroups.

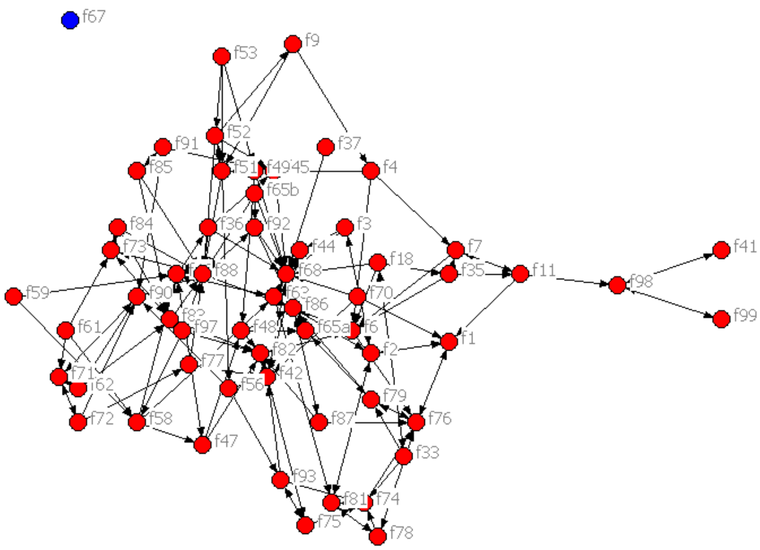


Figure 7.7: NetDraw Visualization of the Attiro Network's Weak Components

Identifying Components in Pajek

Net>Components >Strong, Strong-Periodic, Weak, Bi-Components

The *Net>Components* menu in Pajek has a submenu to find three types of components: strong, strong-periodic, weak, and bi-components. As noted earlier, we will only consider strong and weak components, while delaying our discussion of bi-components until the chapter on brokerage and not covering strong-periodic components at all. When you execute the *Strong* or *Weak* command, a dialog box appears that asks for the minimum size of components. Since small components are generally uninteresting, you may want to increase the minimum size to three or higher to eliminate isolated vertices, which are counted as separate components, and dyads (components of size two). The command creates a partition in which each class of actors represents a component (actors who do not belong to a component are assigned class '0'). You can examine the new partition using the "Edit Partition" button (located just to the left of the partition dropdown list) or the *File>Partition>Edit* command. If you draw the network and partition using the *Draw>Draw-Partition* command (make sure the network and strong components partition are highlighted in the network and partition dropdown lists respectively); it should look similar Figure 7.8 (next page) – although the color of the nodes may differ as well as the size (I adjusted the node size to 10 using the *Options>Size>of Vertices* command).

File>Partition>Edit

Draw>Draw-Partition

[Draw Screen] Options>Size>of Vertices

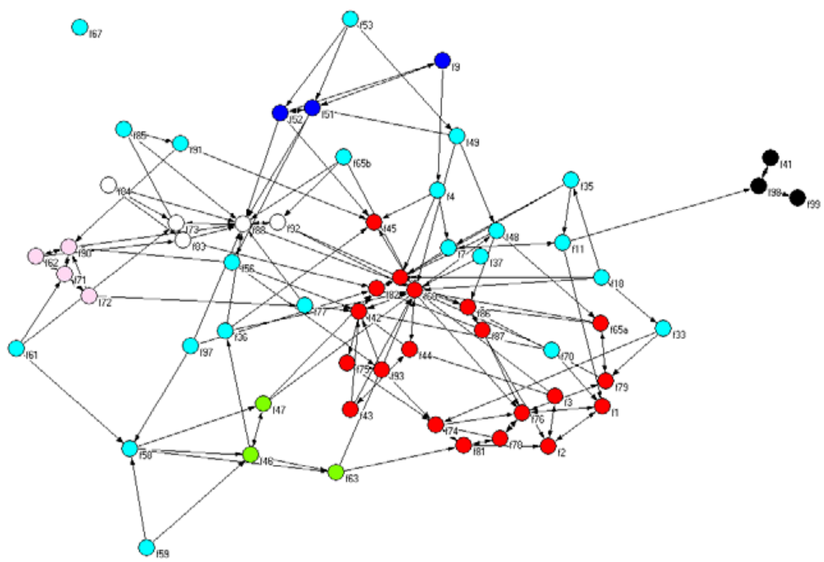


Figure 7.8: Pajek Visualization of the Attiro Network's Strong Components (with actors belonging to no component showing)

Operations>Extract from Network>Partition

Draw >Draw>Partition

Say, you are interested in extracting families that are members of a component. With the network and strong components partition highlighted in their respective dropdown lists at Pajek's main screen, issue the *Operations> Extract from Network>Partition* command. At the dialog box tell Pajek to extract all actors in class one or higher by typing 1-* (not shown); then revisualize the new network (and partition) using Pajek's *Draw>Draw-Partition* command. After reenergizing the network, it should look something like Figure 7.9 below.

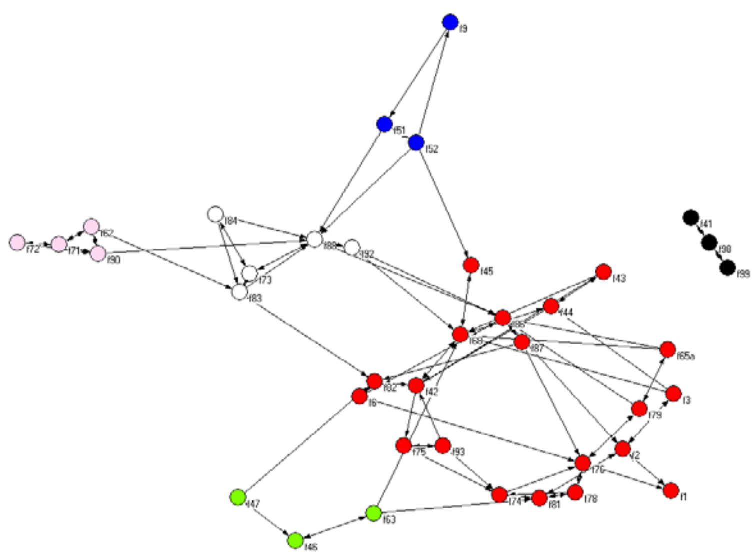
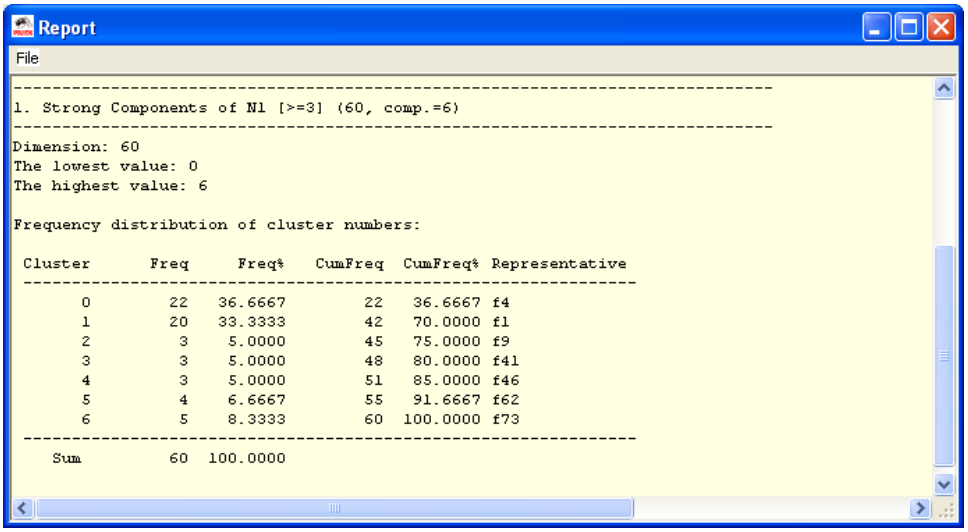


Figure 7.9: Pajek Visualization of the Attiro Network's Strong Components

Info>Partition

Finally, you may want to extract the largest component out of the overall network. In order to do this, you first have to identify it. To do this, make sure that the original strong component partition appears in the partition drop down box and then select the *Info>Partition* command. The resulting report window should look similar to Figure 7.10 below.



```
Report
File
------------------------------------------------------------------------------
1. Strong Components of N1 [>=3] (60, comp.=6)
------------------------------------------------------------------------------
Dimension: 60
The lowest value: 0
The highest value: 6

Frequency distribution of cluster numbers:

 Cluster      Freq      Freq%    CumFreq  CumFreq% Representative
 ----------------------------------------------------------------
       0        22    36.6667         22    36.6667 f4
       1        20    33.3333         42    70.0000 f1
       2         3     5.0000         45    75.0000 f9
       3         3     5.0000         48    80.0000 f41
       4         3     5.0000         51    85.0000 f46
       5         4     6.6667         55    91.6667 f62
       6         5     8.3333         60   100.0000 f73
 ----------------------------------------------------------------
    Sum         60   100.0000
```

Figure 7.10: Pajek Report Window Concerning Partition Information

Operations>Extract from Network>Partition

Draw>Draw

As you can see, cluster (class) 1 is the largest of the six components. To extract this component, make sure that the original Attiro network and strong component partition are highlighted in their respective drop boxes, and then select the *Operations>Extract from Network>Partition* command. At the dialog box tell Pajek to extract all actors in class one by typing 1; then revisualize the resulting network using Pajek's *Draw>Draw* command (see Figure 7.11).

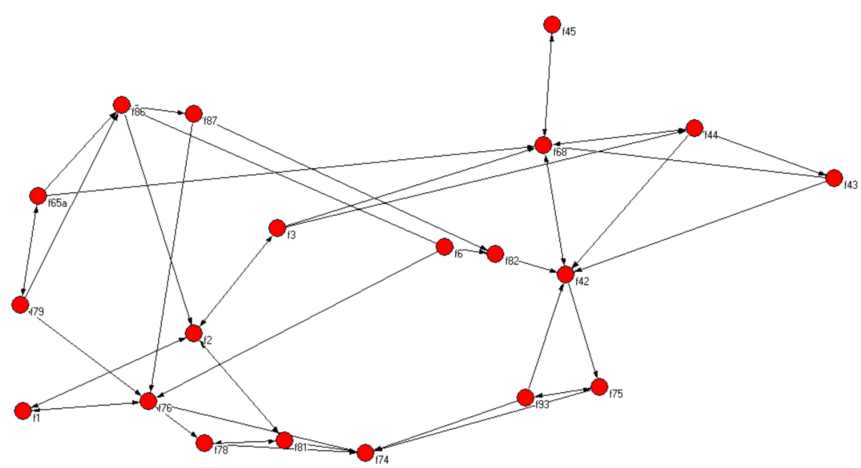


Figure 7.11: Strong Component of Attiro Network

File>Pajek Project File >Read F1

Operations >Extract from Network >Partition

Pajek>Components>Weak

As we saw when examining how UCINET identifies components, the combined Noordin Network is so well-connected, virtually all of the actors in the network belong to the same component. The only exceptions to this were those actors and dyads of actors that were isolated from the larger network. What we did not examine, however, was whether we could identify meaningful components with Noordin's "Alive and Free" network, even though it is an undirected network. Using the Combined Noordin Network project file (remember to read in the project file, not only the network file, using Pajek's *File>Pajek Project File> Read F1* command. Next, after insuring that the "Alive and Free" partition is highlighted in the partition dropdown box, extract the alive and free network using Pajek's *Operations>Extract from Network>Partition* command. Be sure to select cluster (i.e., classes) "1" since "0" = dead or in custody and "1" = alive and free. Next, issue Pajek's *Net>Components>Weak* command (since it is an undirected network), using three as the cutoff value, and Pajek will create a new partition. Visualize the Alive and Free network with the new partition using Pajek's *Draw>Draw-Partition* command. It should look similar to Figure 7.12.

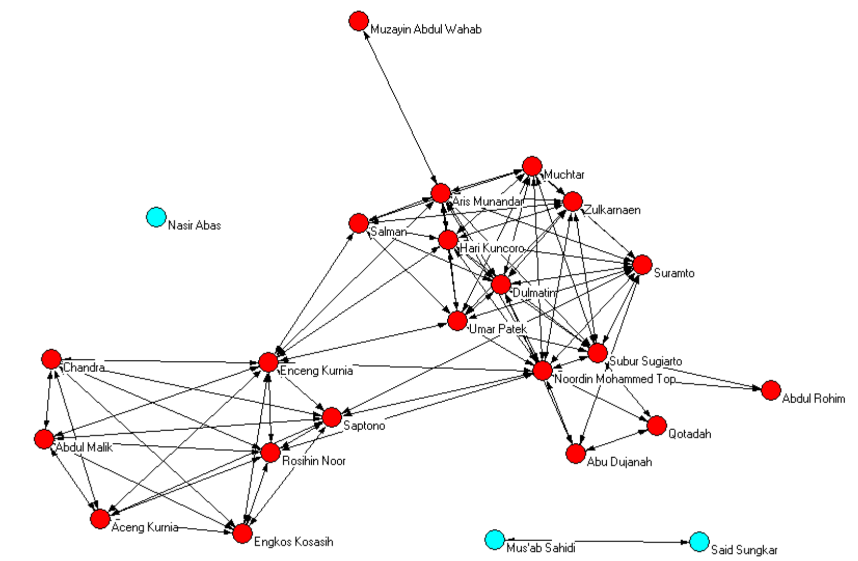


Figure 7.12: Weak Components of Noordin Alive and Free Network

As you can see, in the Alive and Free network there is only one component, even though visually there appears to be two relatively distinct clusters of actors. This is a good example of how social network routines for identifying cohesive subgroups are not always successful, which is why we often have to employ a number of algorithms in our analyses. The next algorithm we will consider is Cores (or k-cores), which builds upon individual actor degree centrality scores. It is to that topic that we now turn.

7.2 Cores

While degree centrality reveals the number of ties to individual actors, it does not indicate whether highly connected actors are clustered or scattered across a network (de Nooy, Mrvar and Batagelj 2005). However, we can use degree centrality to identify clusters of actors that are tightly connected. With this approach we do not pay attention to the degree of one actor but to the degree of all actors within a cluster. Such clusters are called *k*-cores where *k* indicates the minimum degree of each actor within the core. Formally, a *k*-core is a maximal group of actors, all of whom are connected to some number (*k*) of other members of the group (Hanneman and Riddle 2005). Thus, a 2-core contains all actors that have two or more ties to other actors (i.e., all actors with a degree centrality of two or more), a 3-core contains all actors that have three or more ties to other actors, and so on. It is important to note, however, that while *k*-core analysis can help identify relatively dense subnetworks, a particular *k*-core is not necessarily a cohesive subgroup itself (de Nooy, Mrvar and Batagelj 2005), which is why after identifying k-cores, analysts sometimes use one of the components routine discussed above to identify subgroups.

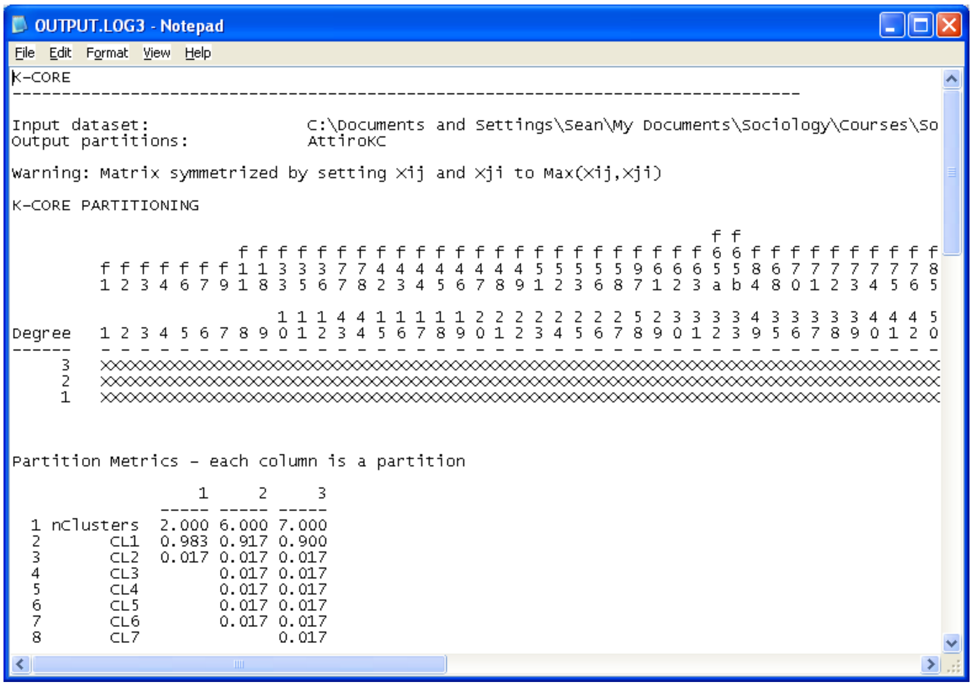
Identifying Cores in UCINET and NetDraw

[UCINET] Network>Regions >K-Cores

In UCINET, *k*-cores are detected using the *K-Cores* command found under the *Network>Regions* submenu, which brings up a dialog box (not shown) that asks you to identify the network you wish to examine. UCINET allows us to change the file name of one output file (in this case the default it KCores), which is somewhat frustrating since the file we will later import into NetDraw for visualization is automatically assigned the name K-Coreness. In other words, we are not given an opportunity to give the file a meaningful name, and (more importantly) it is easy to overwrite the file when conducting multiple k-core analysis.

Here, we begin by analyzing the Attiro network before moving to the Noordin Top dataset. After selecting OK, UCINET produces an output file similar to the one displayed in Figure 7.13 (next page). There are several important things to note about the output. First, UCINET tells you ("Warning) that it symmetrized the network (i.e., it transformed it from a directed to undirected network) and then for each pair of cells, assigned cell values based on the maximum value found in the two cells – in other words, if family A visited family B (value = 1) but family B did not visit family A (value = 0), UCINET assigned a value of 1 (and vice-versa); if neither family visited one another, then

UCINET assigned a value of 0; and, if both families visited one another UCINET assigned a value of 1 (not 2). Next, UCINET's output a K-Core partitioning dendogram that indicates the k-core layers to which each actor belongs. If you scroll across the output file, you will see that, except for family f67 all of the families in the network belong to the 1-core, all but five families belong to the 2-core and all but six families belong to the 3-core. Next, UCINET provides some metrics concerning the proportion of families per k-core, below which you will find a partition that assigns each family to the highest k-core of which it is a part.



```
K-CORE
--------------------------------------------------------------------------------

Input dataset:                 C:\Documents and Settings\Sean\My Documents\Sociology\Courses\So
Output partitions:             AttiroKC

Warning: Matrix symmetrized by setting Xij and Xji to Max(Xij,Xji)

K-CORE PARTITIONING

                                                                 f f
                           f f f f f f f f f f f f f f f f f f f f f f f f 6 6 f f f f f f f f f f
             f f f f f f f 1 1 3 3 3 7 7 4 4 4 4 4 4 4 4 5 5 5 5 5 9 6 6 6 5 5 8 6 7 7 7 7 7 7 7 8
             1 2 3 4 6 7 9 1 8 3 5 6 7 8 2 3 4 5 6 7 8 9 1 2 3 6 8 7 1 2 3 a b 4 8 0 1 2 3 4 5 6 5

                               1 1 1 4 4 1 1 1 1 1 2 2 2 2 2 2 2 2 5 2 3 3 3 3 4 3 3 3 3 3 4 4 4 5
Degree       1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 9 5 6 7 8 9 0 1 2 0
------       - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
     3       XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
     2       XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
     1       XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX


Partition Metrics - each column is a partition

                 1     2     3
             ----- ----- -----
 1 nClusters 2.000 6.000 7.000
 2       CL1 0.983 0.917 0.900
 3       CL2 0.017 0.017 0.017
 4       CL3       0.017 0.017
 5       CL4       0.017 0.017
 6       CL5       0.017 0.017
 7       CL6       0.017 0.017
 8       CL7             0.017
```

Figure 7.13: UCINET K-Core Output File (Attiro Network)

[NetDraw]
File>Open>Ucinet dataset >Network

File>Open>Ucinet dataset >Attribute data

Properties>Nodes>Symbols >Colors>Attribute-based

If we open the Attiro network ("Attiro") and the related k-core partition ("K-Coreness") – which is an attribute (not a network) file – in NetDraw using its *File>Open>Ucinet dataset>Network* and *File>Open>Ucinet dataset>Attribute data* commands respectively and then assign node colors based on the k-core partition using NetDraw's *Properties>Nodes>Symbols>Colors>Attribute-based* command should result in a network map similar to the one shown in Figure 7.14 below (MDS was used to energize the network). One thing worth pointing out is that four families that belong to the same k-core (in blue in the diagram below) are not all tied to one another. Family f37 is disconnected from the other three. Clearly, they do not form a cohesive subgroup. Nevertheless, by extracting the highest k-core from a network (in this case a 3-core), we often can identify cohesive subgroups. We cannot do this in NetDraw, however. We have to return to UCINET, which allows users to extract subnetworks much like Pajek does

(although its extraction capabilities are somewhat limited and not nearly as robust as Pajek's).

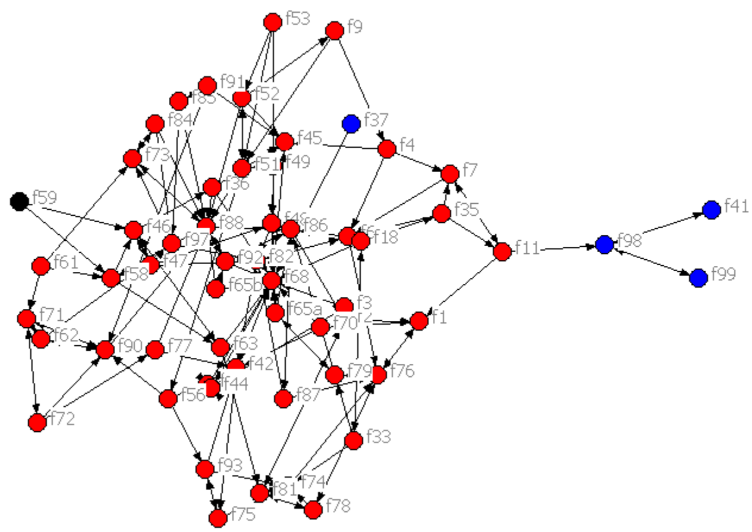


Figure 7.14: Attiro Network K-Cores

[UCINET] Data >Subgraphs from partitions

To extract subnetworks in UCINET we use its *Data>Subgraphs from partitions* command, which brings up the following dialog box (Figure 7.15). As the input network I have chosen the Attiro file and as the input partition I have chosen the K-Coreness file.

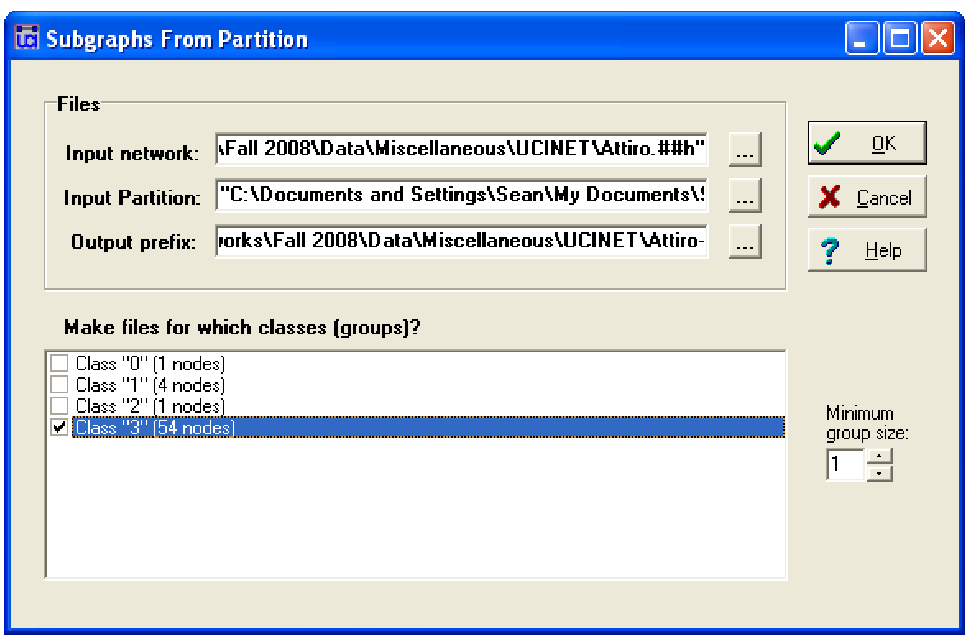


Figure 7.15: UCINET Subgraphs from Partition Dialog Box

As you can see in the lower half of the dialog box, UCINET identifies how many different classes are included in the K-Coreness partition and allows users to identify which partitions they want to extract by checking the box next to the class. In this case, I have only selected the 3-core network to be extracted. If I had wanted to extract the 2-core and 3-core subnetworks into separate networks, I could have done that as well. Unfortunately, UCINET does not allow us to extract multiple classes of networks into a single network; in other words, I could not extract actors belonging to the 2 and 3-core networks into a single network. To do that, we would need to turn to Pajek (see below).[33] After clicking the OK button, UCINET generates a new 54-actor network, consisting of just those actors who belong to the 3-core. Figure 7.16 is a NetDraw drawing of this network.

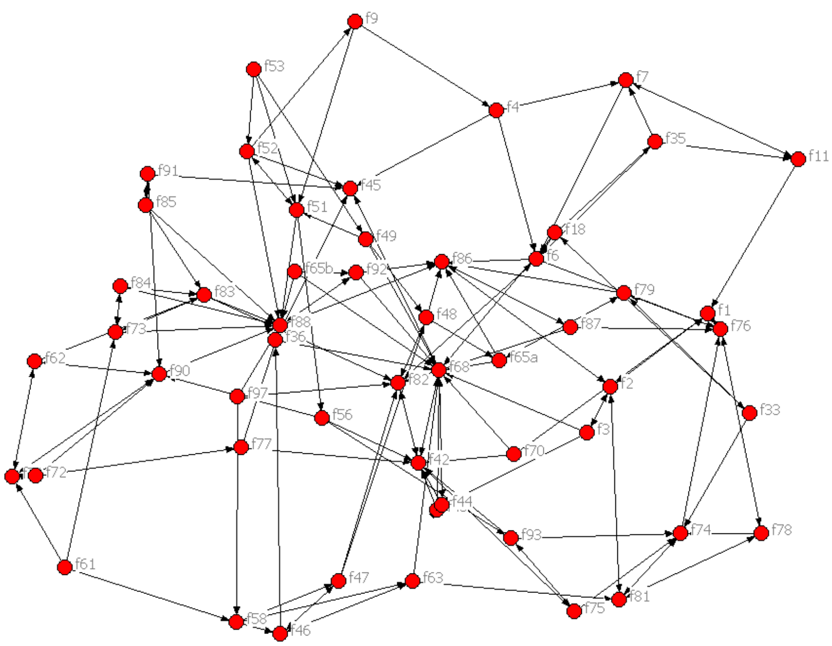


Figure 7.16: Attiro Network 3-Cores

Now let us take a look at the Combined Noordin network. We saw earlier that a weak components analysis of both the complete and alive and free networks did not yield any discernible cohesive subgroups. Perhaps a k-core analysis will produce different results. Figure 7.17 (below) displays a portion of UCINET's output log for the k-core analysis of the Combined Noordin Network. As you can see it identifies 13 different k-cores, ranging from a 2-core to a 34-core. Figure 7.18 (below) is a NetDraw mapping of the same network with the various k-cores

[33] You can identify k-cores in NetDraw by using the *Analysis>K-Cores* command. However, as we saw above, if you want to remove certain core layers you will need to do your data manipulations in either UCINET or Pajek. Nevertheless, NetDraw is a good place to begin your analysis.

indicated by color. As you can see the algorithm appears to identify a central core (34-core, dark grey nodes). Three also appears to be two other distinct clusters, a 25-color (purple nodes) and a 15-core (red nodes) although the latter core does include one disconnected actor.

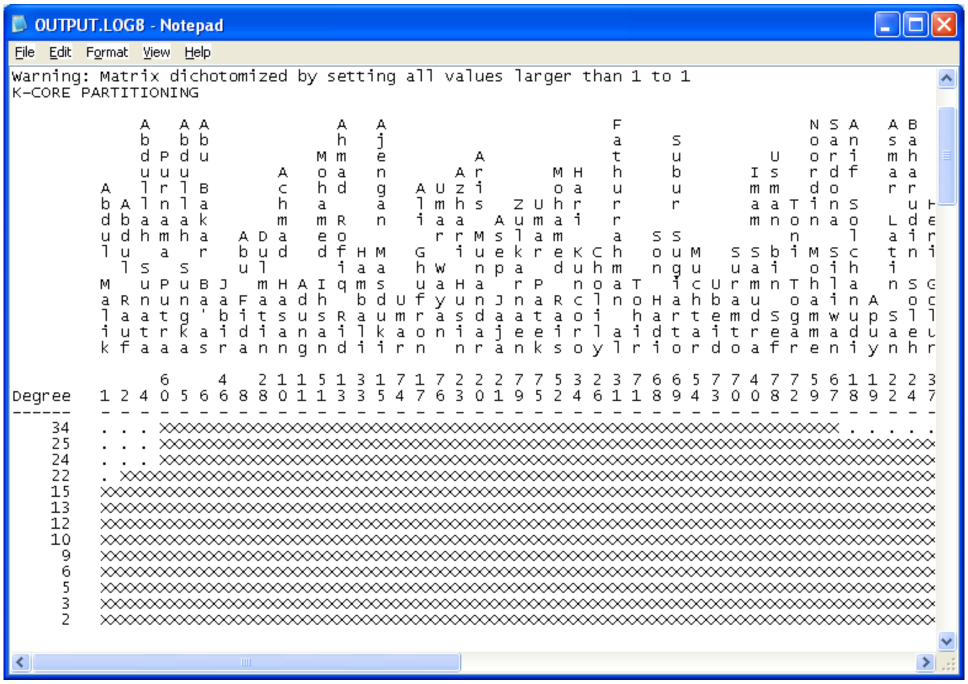


Figure 7.17: UCINET K-Core Output File (Noordin Combined Network)

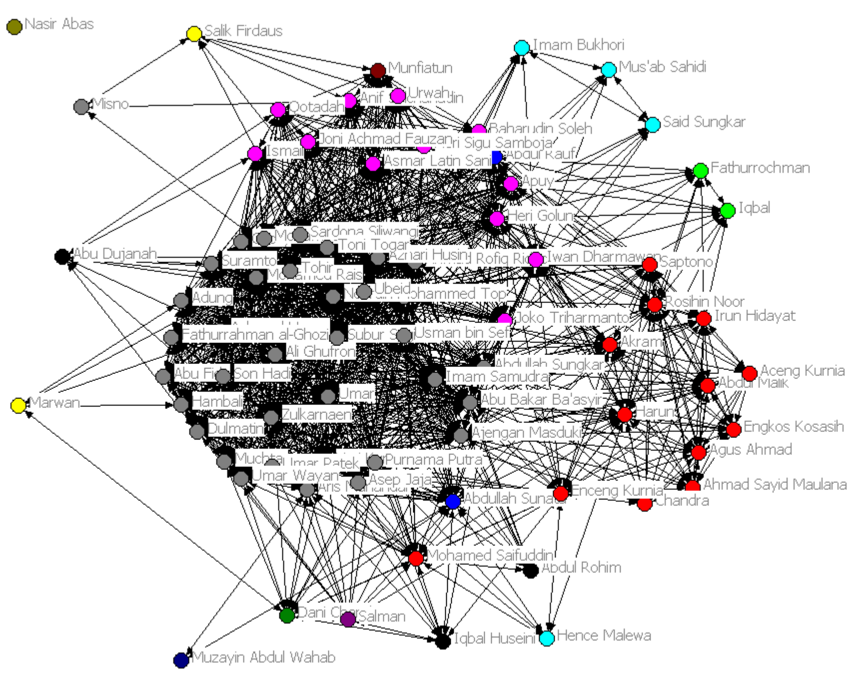


Figure 7.18: K-Cores of Combined Noordin Network

A k-core analysis of Noordin's Alive and Free network yields two, possibly three distinct cohesive subgroups as shown in Figure 7.19 below with one actor (Salman) appearing to function as a broker between the two groups.

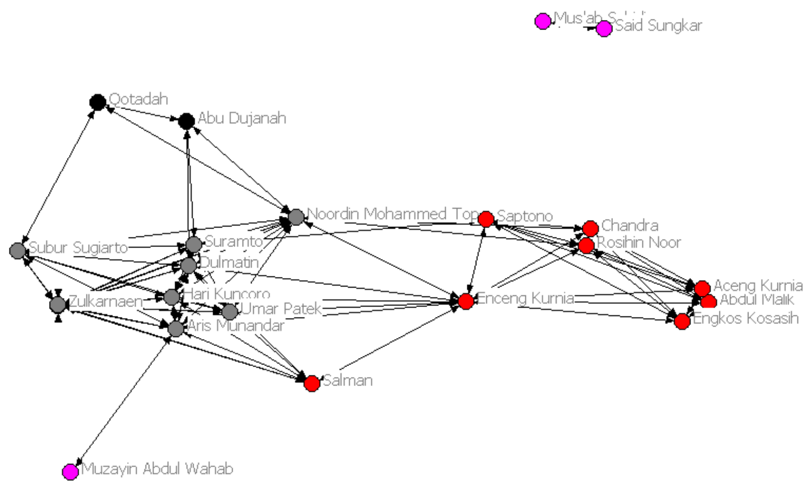


Figure 7.19: K-Cores of Noordin Alive and Free Network

Identifying Cores in Pajek

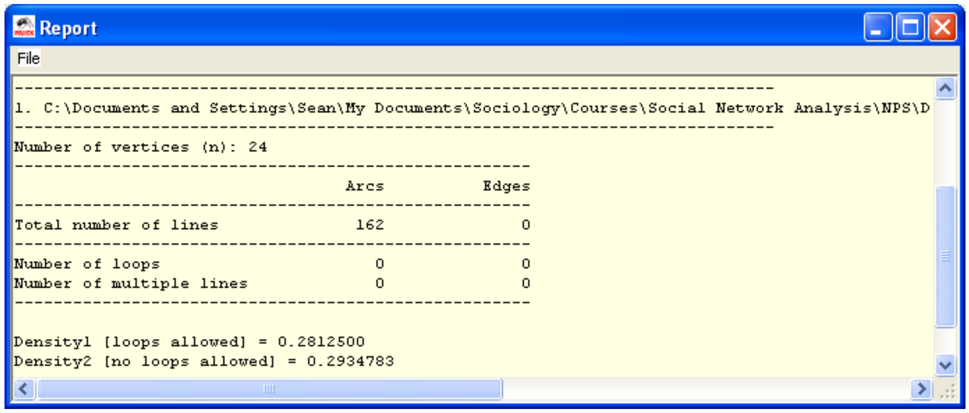
Net>Partitions>Core

In Pajek, *k*-cores are detected with the *Core* command found in the *Net> Partitions* submenu. The *Input*, *Output*, and *All* commands distinguish between input cores (based on the number of lines pointing to a vertex), output cores (based on the number of lines pointing away from a vertex) and cores that ignore the direction of lines. Most of the time you will want to use the *All* command and *only* to apply it to simple undirected networks. The command yields a partition which assigns each vertex to the highest *k*-core in which it appears. Let us try this with the Noordin Alive and Free network.

Info>Network>General

Net>Transform >Arc → Edges>All

After highlighting the Alive and Free network in the network drop down box, check to see whether the network is directed or undirected using the *Info> Network>General* command. This provides a brief output (see Figure 7.20 below) that indicates the number of vertices (actors), lines, loops, and multiple lines in the network. It also indicates whether the lines are arcs, edges or a combination of both. As the report indicates all of the lines in the network consist of arcs, which means that the network is currently a "directed" network, so we need to convert the network into an undirected network by transforming all of the arcs into edges. In Pajek this is accomplished using its *Net>Transform>Arc → Edges>All* command. This calls up a dialog box (not shown), which asks you whether you want to create a new network (answer yes); Pajek then asks what you want to with multiple lines; choose option 5 (single line) and click OK.



Report

File

```
--------------------------------------------------------------------------------
1. C:\Documents and Settings\Sean\My Documents\Sociology\Courses\Social Network Analysis\NPS\D
--------------------------------------------------------------------------------
Number of vertices (n): 24
-----------------------------------------------------
                                  Arcs          Edges
-----------------------------------------------------
Total number of lines              162              0
-----------------------------------------------------
Number of loops                      0              0
Number of multiple lines             0              0
-----------------------------------------------------

Density1 [loops allowed] = 0.2812500
Density2 [no loops allowed] = 0.2934783
```

Figure 7.20: Pajek Report of General Network Information

Net>Partitions>Core

Now it is time to identify k-cores within the network, using Pajek's *Net> Partitions>Core* command, which creates a new "all core partition." Using the "Edit Partition" button (located just to the left of the partition drop down list) or the *File>Partition>Edit* command examine the new partition. Note that two k-cores emerge from this analysis – a 6-core and an 8-core. Visualize the Alive and Free network along with this partition, using the *Draw>Draw-Partition* command, which should yield a drawing similar to the one in Figure 7.21.

File>Partition>Edit

Draw>Draw-Partition

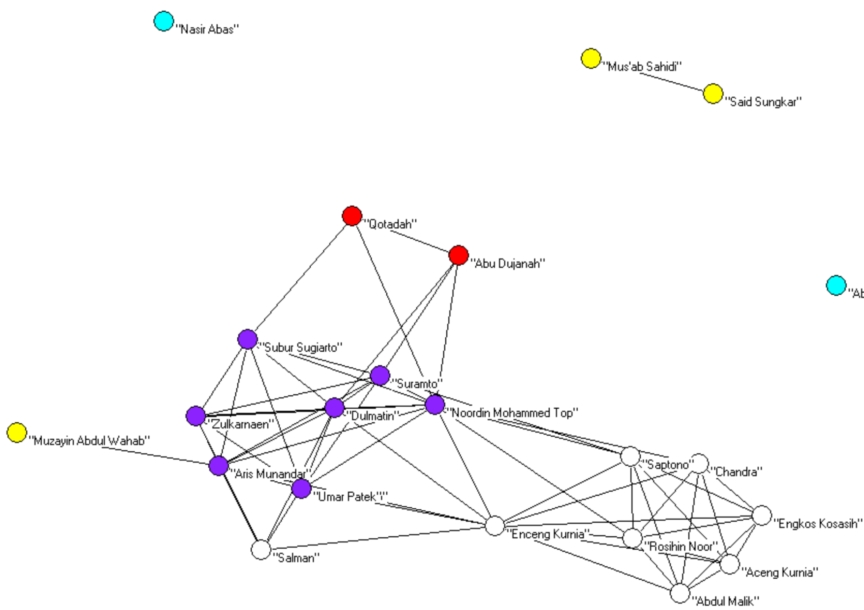


Figure 7.20: Pajek Drawing of Noordin Network K-Cores

Operations >Extract from Network>Partition

With the *k*-core partition you can delete low *k*-cores from the network in order to extract the densest parts in the network. Make sure that the *k*-core partition in the partition drop list and execute the *Operations>Extract from Network>Partition* command. Select the lowest and highest *k*-cores you want to

extract from the network (e.g., 6-8)[34] and redraw the resulting network and partition using the *Draw>Draw-Partition* command, which results in a network map similar to the one displayed in Figure 7.22 below.

Draw>Draw-Partition

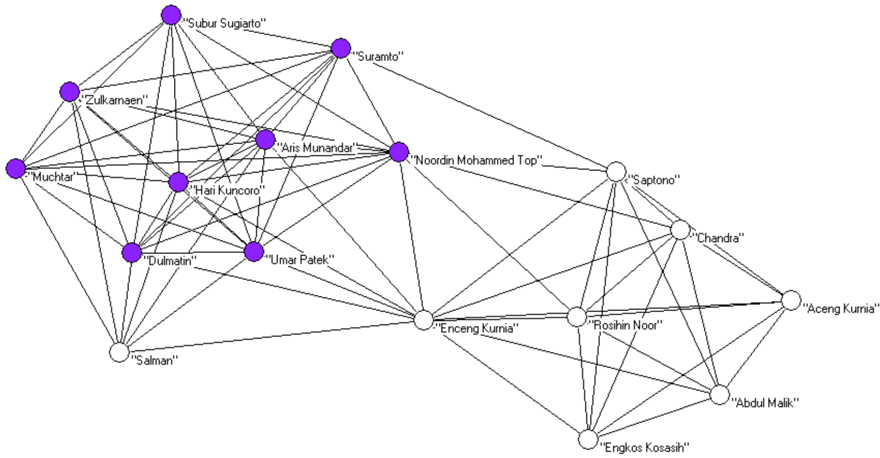


Figure 7.22: Pajek Drawing of Noordin Network 6 and 8-Cores

What is clear that at leas in this instance, a k-core analysis of both the Noordin Combined and Noordin Alive and Free network has proven more fruitful than did the components analysis. That will not always be the case, of course, which again points to the need for analysts to explore networks in a variety of ways in their attempt to get a handle on their underlying structure. Now let us turn to the cohesive subgroup known as cliques.

7.3 Cliques

Perhaps the strictest structural form of a cohesive subgroup is a clique; a clique is a set of vertices in which each actor is directly connected to all other actors. Cliques generally contain a minimum of three actors because although smaller subnetworks exist, they are relatively uninteresting. Take, for example, Figure 7.23 on the next page. Actors 1, 5 & 6 constitute a clique in this network because all three are connected to one another, and we cannot add another actor (e.g., actor 2) that would be connected to all other actors. By contrast actors 2, 4 and 5 are not a clique because we can add actor 3 and the subnetwork would still remain complete (i.e., all four actors would be connected to one another). However, actors 2, 3, 4 & 5 constitute a clique of size four. Detecting cliques in large networks can be a time-consuming process because even medium-sized networks often contain a large number of cliques. Indeed, sometimes you will

[34] Recall that this could not be done in UCINET or NetDraw

find more cliques than actors! Because Pajek's clique-detection facilities are somewhat limited at present, we will only focus on detecting and visualizing cliques using UCINET and NetDraw.

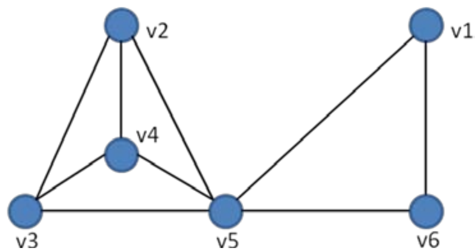


Figure 7.23: Clique example (see de Nooy, Mrvar and Batagelj 2005:66)

Identifying Cliques in UCINET and NetDraw

The UCINET clique algorithm produces a census of all cliques and potentially useful analysis and output. For example, it creates one file (default name = "CliqueSets") that is an affiliation (two-mode) network consisting of actors (rows) and cliques (columns), which can be opened and visualized as a two-mode matrix in NetDraw. It also generates two one-mode networks derived from the "CliqueSets" affiliation network: a "CliqueOverlap," network, which is a co-membership network where ties between actors indicate affiliation with one or more cliques, and the "Clique-by-cliqueOverlap" network, which is a co-event network where ties between cliques indicate that they share one or more actors.

[UCINET] Network>Subgroups >Cliques

First let us look at how UCINET identifies cliques in the Attiro network before moving to the Combined Noordin Network. Select the *Cliques* option under the *Network>Subgroups* submenu. Set the minimum size of cliques at 3, accept UCINET's default options and click OK. This generates an output file (Figure 7.24 below) that, warns the user that although the Attiro network is a directed network, it has treated it as undirected (i.e., "direction of arcs ignored"). Next, it lists the cliques that UCINET identified. As you can see, the Attiro network contains 30 cliques of size three or more, with most of the cliques being of size three. Note that most of the cliques are of size three. Only two are larger than that, and they are only of size four. If you scroll down the output window, you will find an "Actor-by-Actor Clique Co-Membership Matrix" that an examination of the diagonal can tell you which actors belong to the most cliques. In the Attiro network, for example, one family (f51) belongs to four cliques while two others (f68 and f88) belong to five. It is not that surprising that the latter two families are members of numerous cliques since both also score high in terms of degree centrality; however, family f51 is something of a surprise since it is not

ranked in the top five in terms of degree centrality (nor does it score high in terms of closeness or betweenness centrality. Thus, clique analysis may help us identify key actors that other algorithms do not.

```
OUTPUT.LOG6 - Notepad
File  Edit  Format  View  Help
CLIQUES
--------------------------------------------------------------------------------

Minimum Set Size:              3
Input dataset:                 C:\Documents and Settings\Sean\My Documents\Sociology\Courses\Soc

WARNING: Directed graph. Direction of arcs ignored.
30 cliques found.

   1:  f42 f43 f44 f68
   2:  f42 f68 f70
   3:  f36 f45 f68
   4:  f3 f44 f68
   5:  f65b f68 f92
   6:  f4 f6 f7
   7:  f6 f18 f35
   8:  f6 f7 f35
   9:  f7 f11 f35
  10:  f9 f51 f52
  11:  f46 f47 f58
  12:  f46 f58 f59
  13:  f46 f58 f63
  14:  f47 f48 f82
  15:  f48 f65a f86
  16:  f51 f52 f53
  17:  f51 f52 f88
  18:  f49 f51 f53
  19:  f42 f56 f93
  20:  f62 f71 f90
  21:  f71 f72 f90
  22:  f73 f83 f84 f88
  23:  f74 f76 f78
  24:  f74 f78 f81
  25:  f74 f75 f93
  26:  f42 f75 f93
  27:  f65a f79 f86
  28:  f83 f85 f88
  29:  f86 f88 f92
  30:  f65b f88 f92
```

Figure 7.24: UCINET's Listing Cliques Detected in the Attiro network

[NetDraw] Network>Subgroups >Cliques

We can expand our analysis by opening and visualizing the affiliation network generated by UCINET (default = CliqueSets) in NetDraw (remember, it is a two-mode network). Figure 7.25 (next page) is the result. As you can see of the three families that are members of the most cliques (circled in the drawing), family f68 appears to be in a more central position within the overall network. It also appears that there are a handful of cliques that are disconnected from the rest of the group (located in the center of the graph). Indeed, they appear to form a separate subgroup in and of themselves.

What does a clique analysis of the Noordin network yield? First, UCINET tells us that it transformed the network from a valued network to a dichotomized one. Next, it lists the 101 cliques it found in the network (here is an example of the number of cliques outnumber the number of actors!). Visualizing the network in NetDraw (not shown) proves relatively unhelpful in identifying any cohesive subgroups. However, an analysis of the clique-by-clique co-event matrix indicates that there exists within the network, one clique of size 35, which happens to be clique number "1." In order to visualize this clique within the larger network, we need to use UCINET's *Data>Extract>Extract submatrix* command to extract those actors affiliated with clique #1 (which is located in column 1) from the larger "CliqueSets" matrix, as illustrated in Figure 7.26.

[UCINET]
Data>Extract
>Extract submatrix

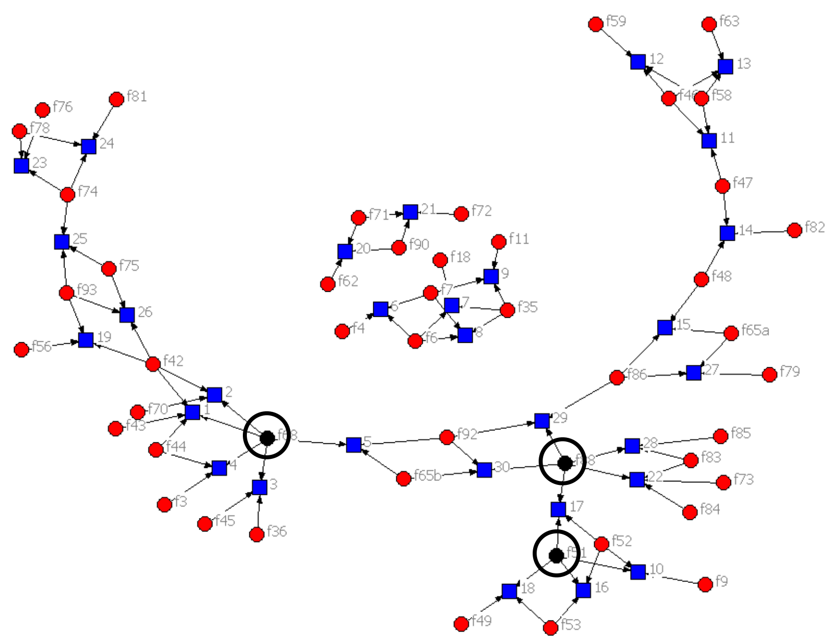


Figure 7.25: Clique Affiliation Network (Attiro)

This effectively creates a partition of 0's and 1's where "1" indicates membership in the clique. Next, in NetDraw open the Combined Noordin Network and the extracted submatrix using NetDraw's *File>Open>Ucinet dataset>Network* and *File> Open>Ucinet dataset>Attribute data* commands respectively. Finally, using NetDraw's *Properties>Nodes>Symbols>Color> Attribute-based* command, adjust the color of the nodes so that they reflect their membership (or non-membership) in clique #1 and you end up with a network map similar to Figure 7.27 (next page).

[NetDraw]
File>Open>Ucinet dataset
>Network

File>Open>Ucinet dataset
>Attribute data

Properties>Nodes>Symbols
>Color>Attribute-based

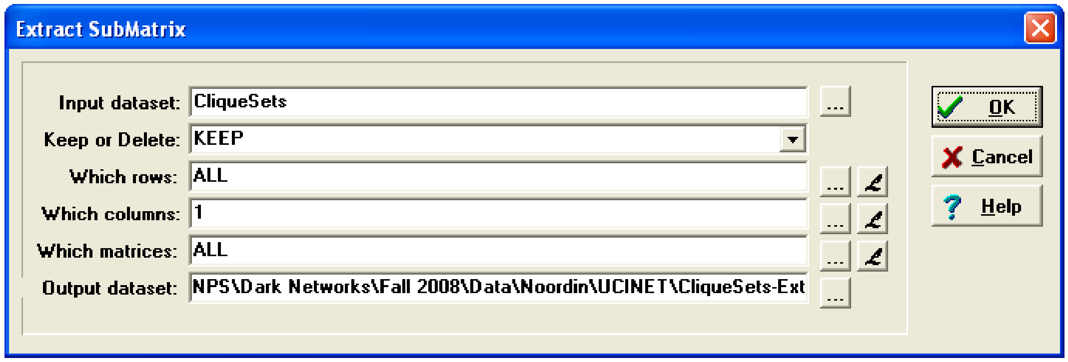


Figure 7.26: UCINET 's Extract Submatrix Dialog Box

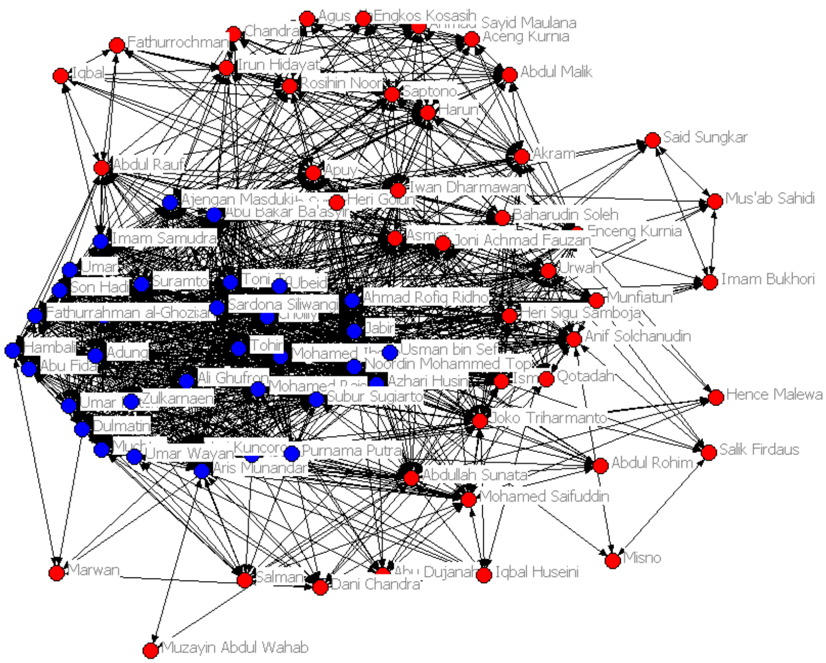


Figure 7.27: Noordin Network's Largest Clique

What might this clique analysis suggest? Well, it appears that within the larger Noordin network, there exists a core of individuals (not core in the k-core sense discussed in the previous section) who are more cohesively tied together than are others in the network. We know from previous research that peripheral members are far less committed and more likely to leave a group than are core members (Popielarz and McPherson 1995; Stark and Bainbridge 1980). Thus, if we were interested in crafting a strategy that utilized an amnesty and reconciliation approach (see e.g.,Mydans 2008), we would probably want to focus our efforts, not on the core members of Noordin's network, but its peripheral members.

N-cliques and N-clans

[UCINET] Network>Subgroups >N-Cliques

[UCINET] Network>Subgroups >N-Clans

Because most of us know "cliques" where some members are not tightly or closely tied to one another, the definition of clique, which requires that every member of a sub-group have a direct tie with each member, strikes many as too strict. UCINET contains other algorithms for indentifying subgroups that relax the "clique" definition somewhat: *N-cliques* and *N-clans*. The *N-clique* approach includes an actor as a member of a clique if they are connected to every other member of the group at a predefined path distance (i.e., '*N*'). In other words, the *N-Clique* approach allows an actor to be a member of a clique even if they do not have ties to all other clique members, just so long as they do have ties to some member, and are no further away than n-steps from all members of the clique. Usually, a path distance of two is used, but larger path distances can be used if need be. You can detect N-Cliques in UCINET using its *Network>Subgroups>N-Cliques* command. Analysts have noted that the *N-Clique* approach sometimes yields stringy groups. To overcome this problem, some researchers restrict *N-Cliques* by requiring that the total span or path distance between any two members of an *N-Clique* also satisfy a condition (e.g., a path distance of two). N-Cliques that satisfy this condition are known as N-Clans, which you can identify in UCINET with its *Network>Subgroups>N-Clans* command. You can also analyze the resulting output from both algorithms in ways similar to how we analyzed the Combined Noordin Network above. Both algorithms also generate two-mode (affiliation) matrices that can then be visualized in NetDraw. The default name for the N-Clique affiliation matrix is "NClqSets," while the default name for the N-Clan two affiliation matrix is "NClanSet."

K-plexes

[UCINET] Network>Subgroups >K-Plex

An alternative way of relaxing the strong assumptions of a pure clique is to allow actors to be members of a clique if they have ties to all but *k*-other members. In other words, the *k-plex* approach assumes that a node is a member of a clique of size *n* if it has direct ties to *n-k* members of that clique. This approach appears to have a lot in common with the *n-clique* approach, but it often yields a very different picture of a network's sub-structures. Rather than the large and "stringy" groups that *n-clique* analysis sometimes produces, *k-plex* analysis tends to find relatively large numbers of smaller groupings. According to Hanneman and Riddle (2005), this tends to focus attention on overlaps and centralization more than on solidarity and reach. UCINET's *K-Plex* command is found under its *Network>Subgroups* submenu.

7.4 Factions

When looking for cohesive subgroups we can imagine a world in which each actor is closely tied to all others in their own subnetwork, but there are no connections at all between subnetworks. If we took all the members of each subnetwork in our ideal world and rearranged their rows and columns so that they were next to each other in an adjacency matrix (i.e. a one-mode matrix), there would be a distinctive pattern of “1-blocks” and “0-blocks” in the resulting matrix. All connections among actors within each subnetwork would be present, while all connections between actors in different subnetworks would be absent (see the “Grouped Adjacency Matrix in Figure 7.28).

```
Group Assignments:

    1:  E F G H
    2:  A B C D

Grouped Adjacency Matrix

        5 6 7 8   1 2 3 4
        E F G H   A B C D
      -------------------
  5 E |   1 1 1 |         |
  6 F | 1   1 1 |         |
  7 G | 1 1   1 |         |
  8 H | 1 1 1   |         |
      -------------------
  1 A |         |   1 1 1 |
  2 B |         | 1   1 1 |
  3 C |         | 1 1   1 |
  4 D |         | 1 1 1   |
      -------------------

Density Table

         1    2
      ---- ----
  1   1.00 0.00
  2   0.00 1.00
```

Figure 7.28: Group Assignments, Grouped Adjacency Matrix & Density Table

[NetDraw] Analysis>Subgroups >Factions

[UCINET] Network>Subgroups >Factions

Not surprisingly, however, most well-connected, undirected social networks do not divide themselves up like this. Nevertheless, the idea of complete connection within and complete disconnection between subnetworks is a useful way for assessing the degree to which a network is “factionalized.” Both NetDraw and UCINET contain a “faction” algorithm that finds the optimal arrangement of actors and measures how well the data actually fit the ideal type.[35] In NetDraw, the command is *Analysis>Subgroups>Factions*, while in UCINET it is *Network>Subgroups>Factions*. Both commands bring up a dialog box that asks you to indicate how many factions (blocks) you would like the algorithm to find. Generally, we use this algorithm in an exploratory way, examining the

[35] Although we do not cover it here, we can conduct factional analysis in Pajek through its block modeling routines (i.e., factions are a special type of block model).

results from several runs with differing numbers of factions and seeing what number yields the best fit (the lower the fitness number, the better the fit).

What does a factional analysis of the Combined Noordin Network produce? When we issue the Factions command in UCINET, it brings up a dialog box similar to the one displayed in Figure 7.29 below. Note that in addition to asking for the input file, UCINET also asks us to indicate how many "blocks" (i.e., subgroups) we wish to identify. Here we may be guided by an a priori hunch as to how many subgroups exist (based on prior analysis), or we may simply engage in an exploratory analysis of the network, varying the number of subgroups, looking for a result that produces the best fit. Before engaging in our analysis, however, we need to first dichotomize the Noordin network using UCINET's *Transform> Dichotomize* command.

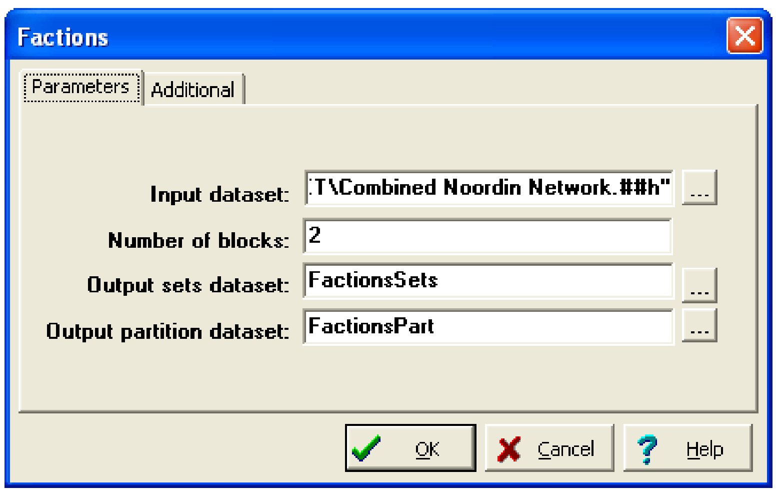
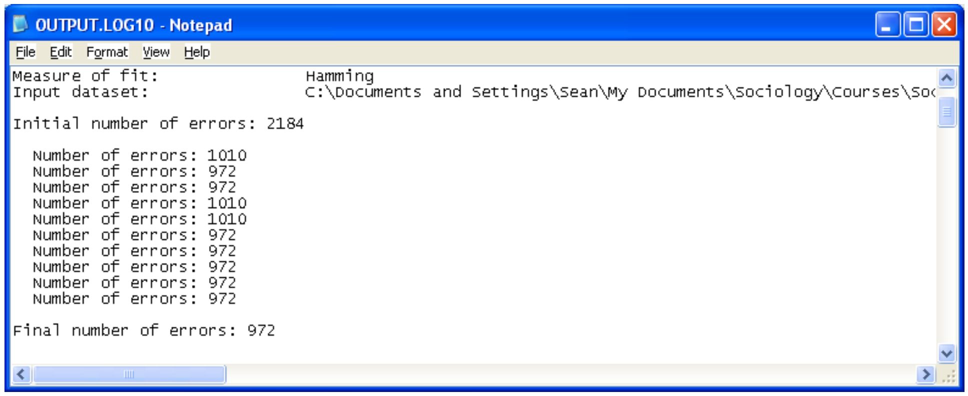


Figure 7.29: UCINET Faction Dialog Box

How do we evaluate the fit of a factional analysis? UCINET provides two metrics: number of errors and density within each subgroup.[36] In factional analysis an error is defined by the absence of a tie between two actors within a subgroup or the presence of a tie between actors located in different subgroups. The number of errors within the network is presented at the beginning of the output file (see Figure 7.30):

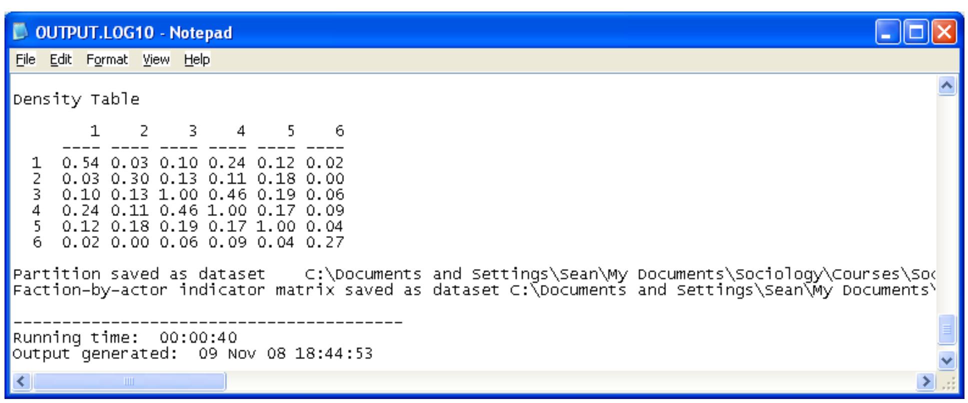
[36] The UCINET output also provides a listing of each member of the various blocks.



```
OUTPUT.LOG10 - Notepad
File Edit Format View Help
Measure of fit:              Hamming
Input dataset:               C:\Documents and Settings\Sean\My Documents\Sociology\Courses\Soc

Initial number of errors: 2184

  Number of errors: 1010
  Number of errors: 972
  Number of errors: 972
  Number of errors: 1010
  Number of errors: 1010
  Number of errors: 972
  Number of errors: 972
  Number of errors: 972
  Number of errors: 972
  Number of errors: 972

Final number of errors: 972
```

Figure 7.30: UCINET Factional Output Listing Number of Errors

Subnetwork density is calculated in the same way that density for the overall network is: total number of ties divided by the total possible number of ties. A density table is presented at the end of the UCINET output file (see Figure 7.31). In general, when conducting a factional analysis of a network, you want the density of each subgroup/block (found along the diagonal of the output) to be greater than the network's overall density. I ran 2, 3, 4, 5 and 6-block factional analyses of a dichotomized version of the Combined Noordin Network. The six-block yielded the lowest number of errors (972) with the density of four of six blocks (Figure 7.31) greater than the network's overall density (.3927), while a five-block analysis yielded a slightly higher error (1008) with the density of four of five blocks greater than the network's overall density. What is striking about the six-block analysis is that three of the blocks (3, 4, and 5) have a density of 1.00, so I chose the six-block.



```
OUTPUT.LOG10 - Notepad
File Edit Format View Help

Density Table

        1    2    3    4    5    6
     ---- ---- ---- ---- ---- ----
  1  0.54 0.03 0.10 0.24 0.12 0.02
  2  0.03 0.30 0.13 0.11 0.18 0.00
  3  0.10 0.13 1.00 0.46 0.19 0.06
  4  0.24 0.11 0.46 1.00 0.17 0.09
  5  0.12 0.18 0.19 0.17 1.00 0.04
  6  0.02 0.00 0.06 0.09 0.04 0.27

Partition saved as dataset    C:\Documents and Settings\Sean\My Documents\Sociology\Courses\Soc
Faction-by-actor indicator matrix saved as dataset C:\Documents and Settings\Sean\My Documents\

----------------------------------------
Running time:  00:00:40
Output generated:  09 Nov 08 18:44:53
```

Figure 7.31: UCINET Factional Density Table Output

[NetDraw]
File>Open>Ucinet dataset >Network

File>Open>Ucinet dataset >Attribute data

We can visualize the factional analysis in NetDraw by opening the Combined Noordin Network and the faction partition (default = FactionPart) using NetDraw's *File>Open>Ucinet dataset>Network* and *File>Open>Ucinet dataset>Attribute data* commands respectively (Figure 7.31).

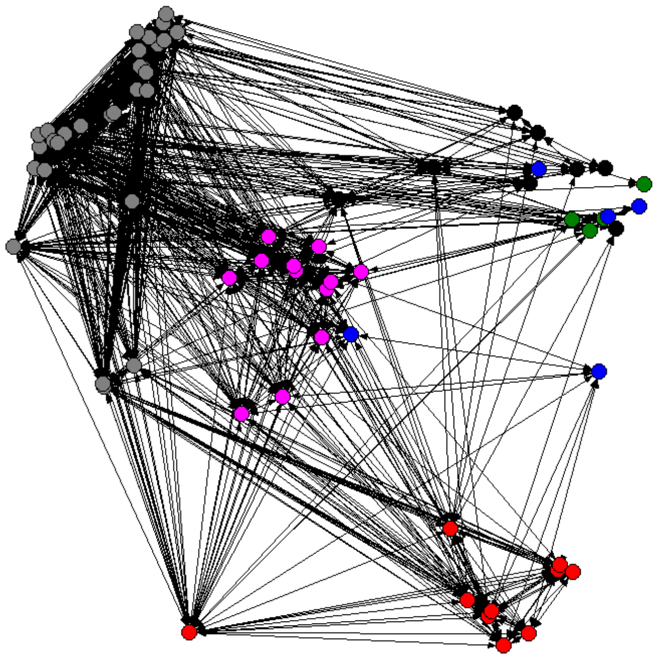


Figure 7.32: NetDraw Drawing of Noordin Factions (Labels Removed)

The network map displayed in Figure 7.32, which was energized using principal components analysis (principal components analysis tends to match up quite nicely with factional analysis), suggests the presence of at least three distinct subgroups (i.e., red, purple and grey groups) within the larger Noordin network. A possible strategy suggested by this analysis might be to use a misinformation campaign to breed distrust between these groups so that they direct their energies inward (i.e., upon each other) rather than outward.

[NetDraw]
Analysis>Subgroups >Factions

NetDraw also includes a factional analysis algorithm, which can be found under the *Analysis>Subgroups* submenu. Like UCINET it calls up a dialog box that asks you how many groups you want to detect (Figure 7.33). Clicking on the OK box will yield a similar (but not necessarily identical) result found in UCINET (not shown). Given this fact, plus the fact that NetDraw does not provide density tables, it is generally better to identify factions in UCINET rather than NetDraw.

Figure 7.33: NetDraw Factions Analysis Dialog Box

7.5 Conclusion

In this chapter we have explored four broad algorithms for detecting cohesive subgroups: components, cores, cliques and factions. There are several more we have not considered. What should be clear by now is that we may have to use multiple algorithms before we succeed in detecting cohesive subgroups. Of course, when working with small groups, we can often detect them visually; however, as groups become larger it becomes increasingly necessary to turn to algorithms in order to detect cohesive subgroups. Once such subgroups are detected, we can then use this information for constructing strategies to disrupt the larger network, such as targeting reconciliation campaigns at peripheral members or using misinformation campaigns to breed distrust between two or more subgroups.

CHAPTER 8

BROKERS, BRIDGES AND STRUCTURAL HOLES

Betweenness centrality, which we examined at length in Chapter 6, implicitly introduced the concept of brokerage, namely, the idea that certain actors are more likely to control the flow of information than are others. In this chapter we examine the notion of brokerage in more depth. We begin by looking at Ron Burt's (1992a; 1992b) notion of structural holes, which builds upon Mark Granovetter's (1973; 1974) work on weak ties. Burt argues that actors who sit on either side of bridges (i.e., ties) that span gaps in the social structure (i.e., structural holes) are in a position to broker the flow of resources across such bridges. Conceptually related to Burt's approach is the one that we will explore in the second section of this chapter, which draws on a form of component analysis (known as bi-component analysis) to identify the bridges and actors that need to remain in place in order for a network to remain connected. Implicit in both the structural hole approach and the bi-component approach is that identifying brokers and the ties that bind them is part and parcel with the cohesive subgroups of which they are (and are not) a part. Put differently, both approaches bring together aspects of the previous two chapters (hence why we examine it here and not in previous chapters). The third and final section of this chapter focuses on an algorithm that explicitly brings these two aspects together. It assumes that brokerage is a function of the different groups with which actors are affiliated; thus, not only does require network data, it also requires attribute data indicating the specific groups to which people belong.

8.1 Bridges and Structural Holes

In order to understand Ron Burt's notion of structural holes we need to first briefly recall what Mark Granovetter discovered when exploring how people had acquired their current jobs. He found that they were far more likely to have used personal contacts in finding their present job than by other methods. Moreover, of those who found their jobs through personal contacts, most of these contacts were acquaintances (i.e., weak ties) not close friends (i.e., strong ties). Why was this so? Granovetter argued that because our acquaintances are less likely to be socially involved with one another than are our close friends, they play an important role in terms of the overall structure of a network because they form the crucial bridges that tie together densely knit clusters of people. In fact, if it were

not for these weak ties, these clusters would not be connected at all. Granovetter in fact argued that while not all weak ties are bridges, all bridges are weak ties.

Burt's approach builds on Granovetter's argument but he takes exception to the idea that only weak ties may be bridges. He concedes that weak ties are more likely to form bridges, but he contends that both weak and strong ties can function as bridges. He does this in order to direct attention away from the type of tie and toward the gaps in the social structure they span. He calls these gaps in the social structure, "structural holes," and he argues that individuals whose ties span these gaps, regardless of whether they are weak or strong ties, are at a competitive advantage over those whose ties do not because such ties provide actors with the opportunity to broker the flow of resources.

In constructing his structural holes measure, Burt focuses on actor's ego networks (i.e., the actor, their neighbors, and the ties between them), the triads in which they are embedded, and the constraint (or lack thereof) their position in these triads places on them. To get a sense of this compare the three different types of triads pictured below (Figure 8.1), which consists of three actors (ego, alter and other) and the ties between them. According to Georg Simmel, when three people are fully connected, such as in the first triad, they share norms, create trust, and manage conflicts (de Nooy, Mrvar and Batagelj 2005:144). However, in the other two incomplete triads, actors in the middle are at an advantage because they are in a position to broker between the other two. Moreover, in complete triads actors cannot withdraw from either tie without creating a structural hole around them. In the first triad "ego" is essentially forced to maintain ties to both of the actors in the triad in order to prevent the formation of a structural hole. For instance, if he were to cut his tie to the third actor, then a structural hole will form between the two that "alter" could then exploit to her advantage.

Figure 8.1: Three Triads

In short, while incomplete triads provide brokerage opportunities for some actors, complete triads offer only constraints, which is why Burt's structural holes measure does not identify structural holes per se but rather estimates the constraint that all actors in a network face in light of the triads in which they are embedded. Less constraint, of course, indicates more brokerage potential. At this point we do

not have to explore the intricacies of how Burt calculates constraint,[37] but the following example (Figure 8.2) taken from de Nooy et al (2005:146) should provide a basic understanding of the assumptions lying behind it. Let us first consider Alejandro's tie to Bob. It is characterized by three incomplete triads because Domingo, Carlos and Eduardo are not directly connected to Bob, which provides Alejandro with the opportunity to broker between Bob and Domingo, Bob and Carlos, and Bob and Eduardo. By contrast, the constraint placed on Alejandro's ties with Carlos, Domingo, and Eduardo is quite high because all of these ties are involved in complete triads. Thus, if Alejandro were to withdraw from any of these relations, he would provide brokering opportunities for one or more of his neighbors. Nevertheless, if we assume that Carlos, Domingo and Eduardo have no other ties, then they face more constraint (and less brokerage potential) than does Alejandro because Alejandro is embedded in incomplete triads while Carlos, Domingo and Eduardo. Basically, then, Burt's structural holes algorithm calculates the level constraint that each triad places on an actor, weights this constraint by the number of ties in an actor is involved and then sums the resulting calculations to arrive at a measure of aggregate constraint. If you subtract this calculation from one, then you arrive at a measure of brokerage potential.

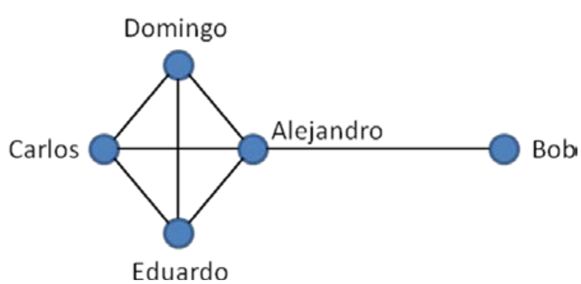


Figure 8.2: Alejandro's Ego-Network (de Nooy, Mrvar and Batagelj 2005:146)

We will first examine how UCINET and NetDraw implement Burt's structural holes measure before turning to Pajek. All three programs calculate the aggregate constraint for each actor, but only NetDraw provides the additive inverse of the constraint measure (i.e., 1 minus aggregate constraint). It is easy to calculate in Pajek although it does take an additional step. We will also see that there is a mathematical problem with Burt's formula when isolates are present in a network. Isolates are accorded low levels of constraint (and thus high brokerage potential), which makes no sense because it is hard to imagine how an actor with

[37] We will explore some of these details as we examine how UCINET and Pajek calculate actor constraint and brokerage potential.

no ties is in a position to broker anything. In UCINET and NetDraw we have to adjust for this, while in Pajek we do not because it makes the adjustment on its own.

Constraint (Structural Holes) in UCINET & NetDraw

We will begin by analyzing a communication network ("Strike.##h" and "Strike.paj") of a wood-processing facility used by de Nooy et al (2005) and available at the Pajek website. The facts were these: a new management team proposed changes to the workers' compensation package, which the workers did not accept and eventually led them to strike. After weeks of failed negotiations, management asked an outside consultant to analyze the communication structure among the employees because it felt that information about the proposed changes was not being effectively communicated to all employees by the union negotiators. The outside consultant asked all employees to indicate the frequency with which they discussed the strike with each of their colleagues. He used a 5-point scale, ranging from 'almost never' (less then once per week) to 'very often' (several times per day). If at least one of two persons indicated that they discussed work with a frequency of three or more, he assumed that a tie existed between the two employees (i.e., a line was drawn between them in the informal communication network). Figure 8.3 presents a drawing of the network (notice that the actors present in Figure 8.2 are represented here as well).

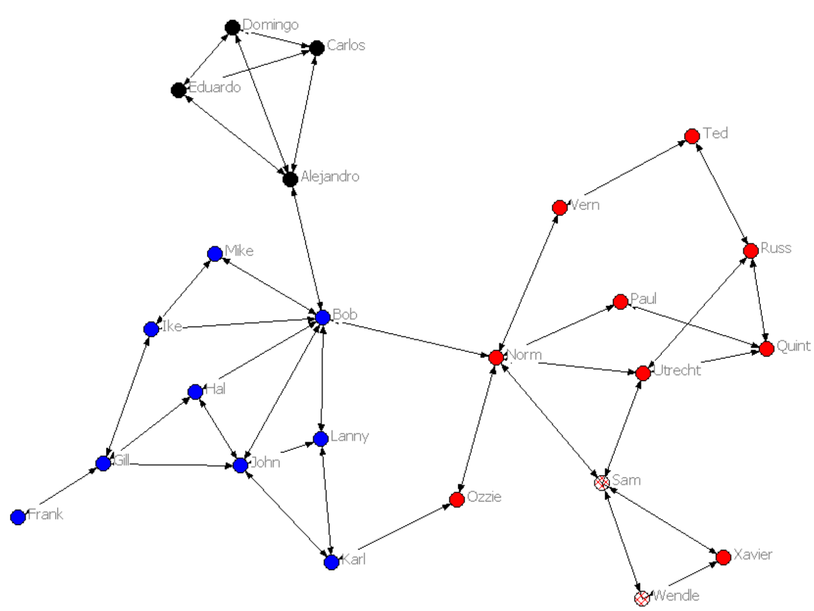


Figure 8.3: Strike Network

The network map displays distinct separations between groups, which are defined by age and primary language. The Spanish-speaking young employees (in the upper left corner of the graph), who are 30 years or younger, are almost completely disconnected from the younger English-speaking employees (lower left of graph) and have no direct ties to the older English speaking employees (right side of graph). If it were not for the fact that Bob spoke some Spanish and Alejandro spoke English, the Spanish-speaking employees would be completely isolated from the rest of the employees. The younger English-speaking employees are virtually a group amongst themselves as well. Indeed, if it were not for Bob (who owes Norm for getting him his job) and Karl (who is Ozzie's son), they would be completely cutoff from the older English-speaking employees. Interestingly, Sam and Wendle (lower right corner of graph) are the union negotiators, which, as we will see, helps explain why management experienced such difficulty in implementing the new workers' compensation package.

[UCINET]
Network>Ego Networks
>Structural Holes

UCINET calculates the structural hole (constraint) measure with its *Network>Ego Networks>Structural Holes* command. In the dialog box that this command calls up (Figure 8.4) select the *"Whole network model – normal method"* option and click OK. The output log (not shown) first lists measures of dyadic redundancy and dyadic constraint, which are used in calculating aggregate constraint, but the details of which need not concern us here.[38] The output next lists Burt's constraint measure (column three) along with several other measures.

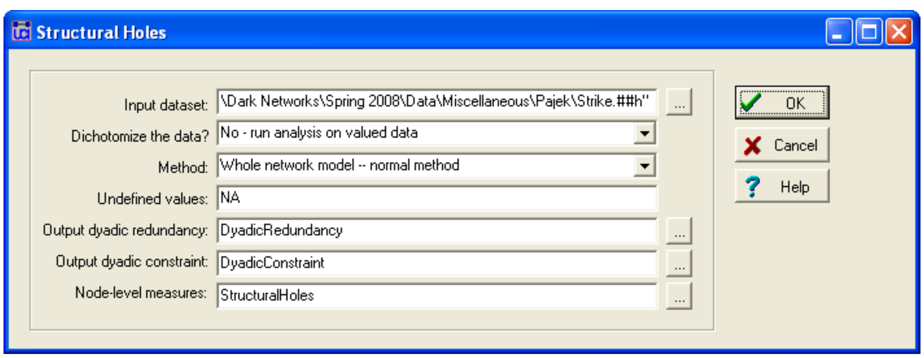


Figure 8.4: UCINET's Structural Holes Dialog Box

[NetDraw]
File>Open>Ucinet dataset
>Network

File>Open>Ucinet dataset
>Attribute data

Next, open the strike data (i.e., Strike.##h) in NetDraw. Also open the Strike_groups.##h and "Structural Holes.##h" attribute datasets using NetDraw's *File>Open>Ucinet dataset>Attribute data* command. Let us vary the color of the nodes to reflect the various groups to which the actors belong and the size of the nodes to reflect the brokerage potential each actor enjoys. To do these two things we need to use NetDraw's *Properties>Nodes>Symbols>Color>Attribute-based*

[38] Hanneman and Riddle (2005:Chapter 9) provide a nice summary of all of UCINET's output.

[NetDraw] Properties>Nodes>Symbols >Color>Attribute-based

Properties>Nodes>Symbols >Size>Attribute-based

(choosing the strike groups attribute) and *Properties>Nodes>Symbols>Size> Attribute-based* commands respectively. Note that in the dialog box that the latter command calls up, we need to select the “Constraint” attribute and check the “Reverse attribute values” box (see Figure 8.5 below), which causes NetDraw to draw the size of the nodes in terms of the inverse of Burt’s measure of constraint. This should produce a network map similar to Figure 8.6 below. As you can see from the drawing, Bob, Norm and Gill enjoy the highest brokerage potential although Gill’s location on the periphery of the network suggests that his brokerage potential is not as robust as are Bob and Norm’s.

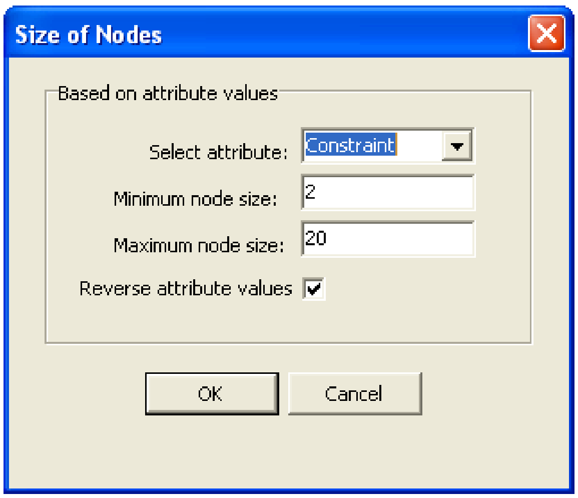


Figure 8.5: NetDraw’s Size of Nodes Dialog Box

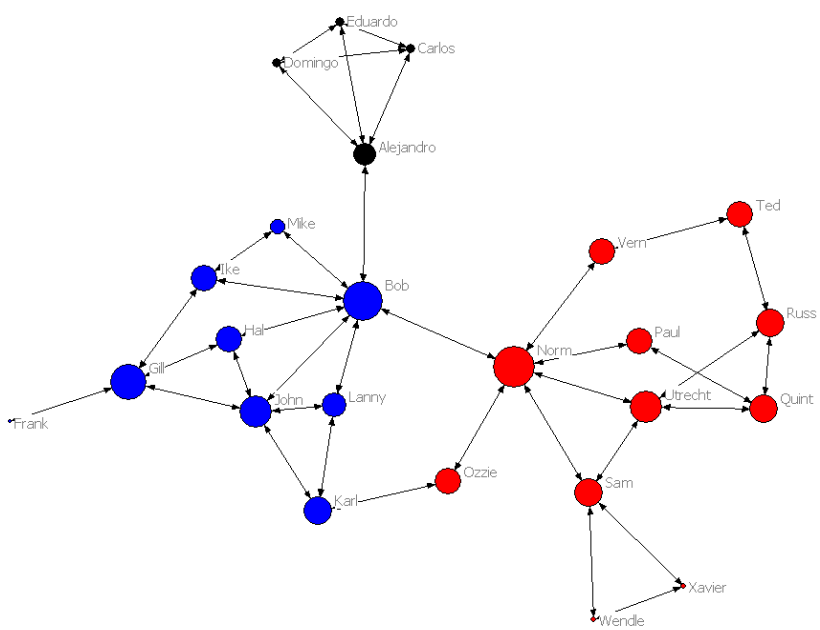


Figure 8.6: NetDraw Map of the Strike Network’s Structural Holes

Turning to the Noordin Alive and Free network (a preliminary structural holes analysis of the Combined Noordin Network did not yield any substantive results), a structural hole analysis (using the same commands as above) yields the following network map (Figure 8.7). Here I have chosen to visualize the dyadic constraint network so that the gaps between actors exhibiting lower constraint are larger. While certain actors (e.g., Noordin Top and Enceng Kurnia) appear to enjoy more freedom to broker the flow of resources than do others in the network, it is not obvious that they do. This suggests that alternative measures of brokerage may be more appropriately used here. What is perhaps more striking, however, is that the size of Nasir Abas's node is larger than those of any other actor in the network even though they are disconnected from the network (and each other). As noted earlier, there is a mathematical problem with Burt's formula when isolates are present in a network. They are accorded low levels of constraint (and thus high brokerage potential) because they are not embedded in any complete triads. Of course, this makes no sense because actors with no ties are hardly in a position to broker any type of resource. One way of addressing this problem is to delete any isolates in the network. Another is to manually assign all isolated actors a constraint measure of one.

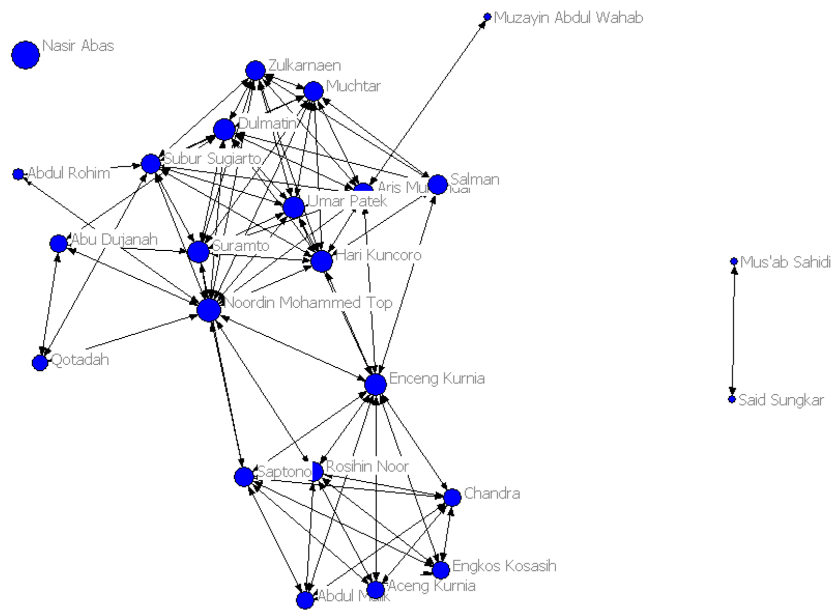


Figure 8.7: Map of the Noordin Alive and Free Network's Structural Holes

Analysis>Structural Holes >Whole Network model

NetDraw provides for the calculation of structural holes as well with its *Analysis>Structural Holes>Whole Network model* command. It also generates a *rConstraint* attribute (i.e., the additive inverse of *Constraint*) that you can use to

visualize your network without checking the "Reverse attribute values" box in the Size of Nodes dialog box.

Constraint (Structural Holes) in Pajek

[Pajek Main Screen]
Net>Vector
>Structural Holes

Draw>Draw

[Pajek Draw Screen]
Layout>Energy
>Kamada-Kawai>Free

Options>Values of Lines
>Similarities

In Pajek the command *Net>Vector>Structural Holes* computes the proportional strength, dyadic constraint, and aggregate constraint for all the actors in a network. Proportional strength and dyadic constraint are output as new networks where the line values express the strength and constraint on relations respectively.[39] To get a sense of the network's structural holes draw the network of dyadic constraint (rather than the original network) and energize it using *Kamada-Kawai>Free*, making sure that the *Similarities* option is selected under the *Options>Value of Lines* submenu found in the *Draw* screen. When you use this network rather than the original network, actors that are tied by relations characterized by high constraint are drawn closely together, whereas relations characterized by low constraint are drawn farther apart, which creates a lot of space between them that looks something like a hole or gap in the social structure.

[Pajek Main Screen]
Info>Vector

File>Vector>Edit

[Pajek Draw Screen]
Options>Size>of Vertices

[Pajek Main Screen]
Draw
>Draw-Partition-Vector

[Pajek Main Screen]
Vector>Transform
>Add Constant

Draw
>Draw-Partition-Vector

Aggregate constraint is output as a vector. Information concerning the vector can be found using the *Info>Vector* command. You can also "inspect" it by selecting the *Edit* option under the *File>Vector* submenu. If you want the size of the vertices in your drawing to represent their respective aggregate constraint, make sure that the *Autosize* option in the *Options>Size>of Vertices* submenu of the *Draw* screen is selected, otherwise the vertices will be drawn too small. Draw the network choosing the *Draw-Partition-Vector* option found under the *Draw* menu in Pajek's main screen. Be sure that the Aggregate constraint vector is showing in the vector dropdown list. If you want the size of the vertex to be positively related to the inverse of the aggregate constraint (i.e., you want larger vertices to indicate better brokerage potential), select the *Vector>Transform>Add Constant* command, and in the dialog box type in the value "-1.00". Then, redraw the network as before, except with the new vector showing in the vector dropdown list. Pajek will inform you that it has drawn negative lines as positive. This is OK since all of the lines are negative. You should get a network map that looks similar to Figure 8.8 below. As before Bob, Norm and Gill enjoy higher brokerage potential than do others in the network, while the ties between Bob and Norm, Bob and Alejandro and Norm and Sam respectively appear to constitute the largest structural holes in the network.

[39] Note that these networks are always directed and that all arcs are reciprocated, no matter whether the original network is directed or undirected, valued or unvalued, whether it contains multiple lines and loops or not.

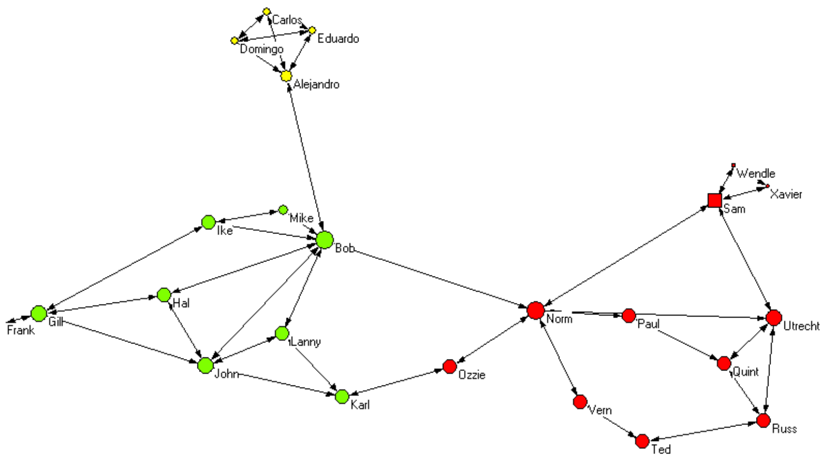


Figure 8.8: Pajek Map of the Strike Network's Structural Holes

[Pajek Main Screen] Info>Vector

File>Vector>Edit

Vector>Transform >Add Constant

Draw>Draw-Vector

If we perform the same analysis on Noordin's Alive and Free network and then examine the aggregate constraint vector, we discover that Pajek has automatically adjusted the aggregate constraint of the network's isolates to the theoretical maximum (i.e., 1.00). After transforming the aggregate constraint vector using the *Vector>Transform>Add Constant* command (adding -1.00) and visualizing the network using the *Draw>Draw-Vector* command, we create a network map similar to Figure 8.9 below. Like Figure 8.7 above none of the actors appear to enjoy a disproportionate advantage in terms of brokerage potential. However, the structural holes between the two clusters are more readily apparent in this network map than in Figure 8.7.

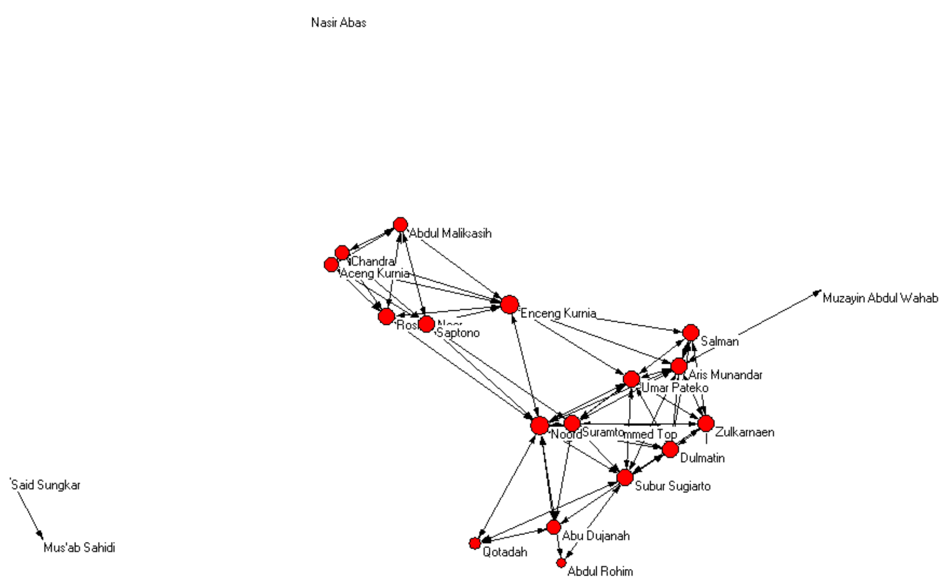


Figure 8.9: Map of the Noordin Alive and Free Network's Structural Holes

8.2 Bridges and Brokers

Reexamining the strike network (Figure 8.10 below) demonstrates why one thing we should always ask about a network is whether there are any bottlenecks that are vital to the flow of resources within the network. The tie between Alejandro and Bob is a bottleneck because removing it will cut the Spanish-speaking employees off from the rest of the employees. Formally, this tie is called a bridge because its removal creates a new component, which is isolated from other components. The tie between Frank and Gill is also a bridge because if you remove it, Frank is isolated from the rest of the network.[40]

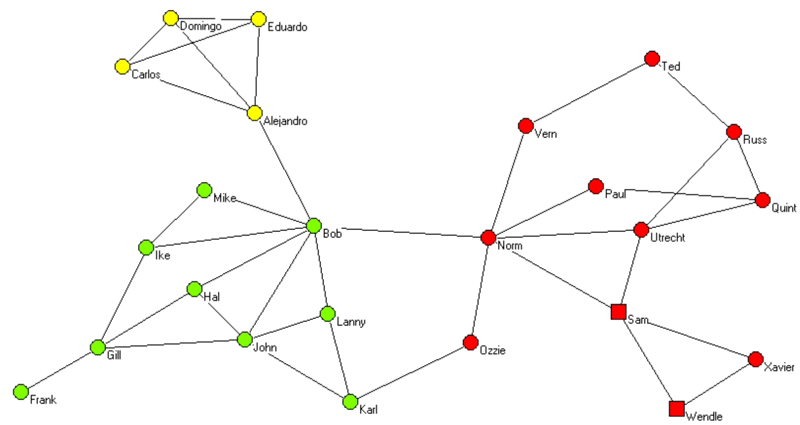


Figure 8.10: Strike Network

Of course, while the removal of a tie can disconnect a network, the deletion of an actor can have the same effect. Actors whose removal disconnect the network or disconnect a component of a network are called cutpoints (UCINET and NetDraw), cut-vertices (Pajek) or articulation points (UCINET and Pajek). Just like bridges, cutpoints are crucial to the flow of resources in a network. In the strike network, Norm is clearly indispensable for the exchange of information between the older and younger employees. Similarly, Alejandro and Bob are necessary in order for an exchange of information between the English-speaking and Spanish-speaking employees to occur. It is important to note that the removal of actors on either side of a bridge eliminate that bridge. Thus, we do not have to necessarily target the more central actor in order to disrupt the flow of resources in a network. Nevertheless, it is also important to point out that actors on either side

[40] Note that this formal definition of a bridge differs from the use of term by Granovetter, who uses the term to refer to any tie that bridges two distinct clusters of actors, regardless of whether they would be completely cut off from one another if the tie was removed. However, here a tie is only

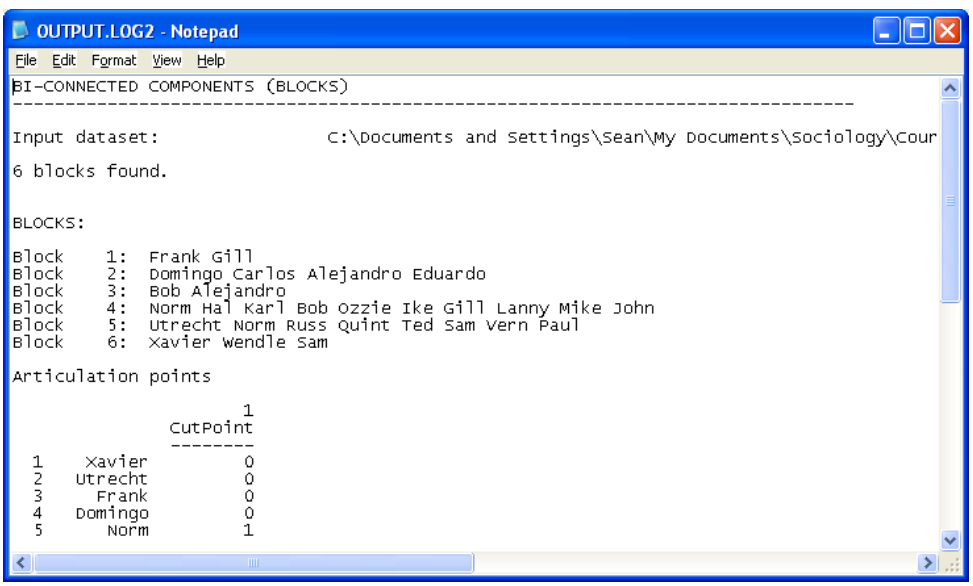
of a bridge may or may not be cutpoints. For example, in the strike network Frank is not one because removing him does not increase the number of components.

What then are bi-components? Simply put, they are components without a cutpoint. They are the sections of a network where the removal of a single actor does not create a new component. This means that in a bi-component no actor completely controls the flow of resources between two other actors because there is always an alternative path that the resources may follow. The strike network, while constituting a single component because information can travel to each employee through the network's various ties, is not a bi-component because the removal of Alejandro, Bob, Sam, Norm or Gill would increase the number of components. However, as the analysis below demonstrates, there are several bi-components embedded in it.

Bridges, Bi-components and Brokers in NetDraw and UCINET

[UCINET] Network>Regions >Bi-Component

In UCINET we detect cutpoints and bi-components using UCINET's *Network>Regions>Bi-Component* command. Unlike Pajek's routine, UCINET does not provide an option for choosing a minimum size of bi-components; instead, it detects all bi-components of all sizes. Thus, when we run the routine on the strike network, UCINET finds six blocks/bi-components (see Figure 8.11).




```
BI-CONNECTED COMPONENTS (BLOCKS)
--------------------------------------------------------------------------------

Input dataset:              C:\Documents and Settings\Sean\My Documents\Sociology\Cour

6 blocks found.


BLOCKS:

Block    1:  Frank Gill
Block    2:  Domingo Carlos Alejandro Eduardo
Block    3:  Bob Alejandro
Block    4:  Norm Hal Karl Bob Ozzie Ike Gill Lanny Mike John
Block    5:  Utrecht Norm Russ Quint Ted Sam Vern Paul
Block    6:  Xavier Wendle Sam

Articulation points

                       1
               CutPoint
               --------
  1    Xavier         0
  2   Utrecht         0
  3     Frank         0
  4   Domingo         0
  5      Norm         1
```

Figure 8.11: UCINET Bi-Component Output Log

---

considered a bridge if its removal would result in the two clusters being cut off from one another(de Nooy, Mrvar and Batagelj 2005).

Generally, however, we are only interested in bi-components of size three or greater since to say that the removal of an actor in a bi-component of size two does not disrupt the flow of information is nonsensical.[41] Thus, in the strike network we find four bi-components of size three or greater: (1) a Spanish-speaking bi-component that includes Alejandro, Carlos, Domingo, and Eduardo, (2) a bi-component that consists all of the young English-speakers (except Frank) plus Ozzie and Norm, (3) one that includes all of the older English-speaking employees except Ozzie, Wendle, and Xavier, and (4) one that consists of just Sam, Wendle and Xavier. What is interesting is that some actors belong to more than one bi-component. Indeed if you take into account bi-components of all sizes in the strike network, the actors who belong to more than one bi-component are Alejandro, Bob, Gill, Norm and Sam, which not coincidentally are also the network's cutpoints.

[NetDraw] Analysis >Blocks & Cutpoints

Because NetDraw's bi-component routine works quite well, there is no need to import the partition files created in UCINET into NetDraw. Instead, use NetDraw's *Analysis>Blocks & Cutpoints* command. This command will color your cutpoints one color and the non-cutpoints another (Figure 8.12).

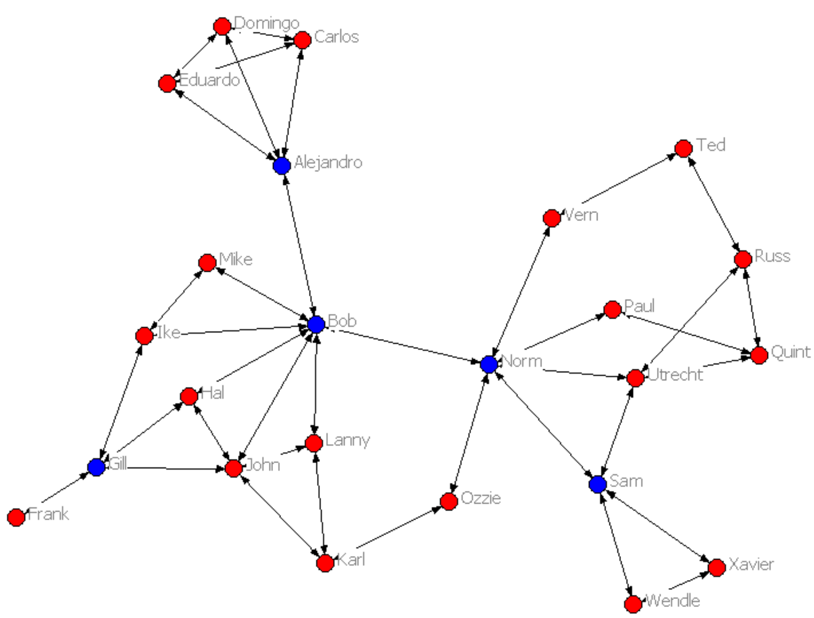


Figure 8.12: Strike Network with Cutpoints Highlighted

[NetDraw] Properties>Nodes>Symbols >Color>Attribute-based

We can change the color of the nodes to reflect the bi-component of which the actors are a part using the *Properties>Nodes>Color>Attribute-based* command, and then in the dialog box, selecting the block (i.e., bi-component) we

[41] de Nooy, Mrvar and Batagelj (2005) call bi-components of size two "bridges."

want to highlight (Figure 8.13 below). If you proceed through each of the blocks, you will see that they match up with those identified in UCINET (which is a good thing).

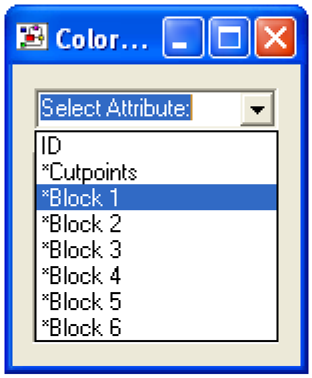


Figure 8.13: NetDraw "Color of Nodes" Dialog Box

Now let us turn our attention to Noordin's network. If you perform a bi-component analysis on the Combined Noordin Network (not shown), you will discover that the entire network constitutes a bi-component and there are no cutpoints. A bi-component analysis of Noordin's Alive and Free network yields similar results (Figure 8.14).

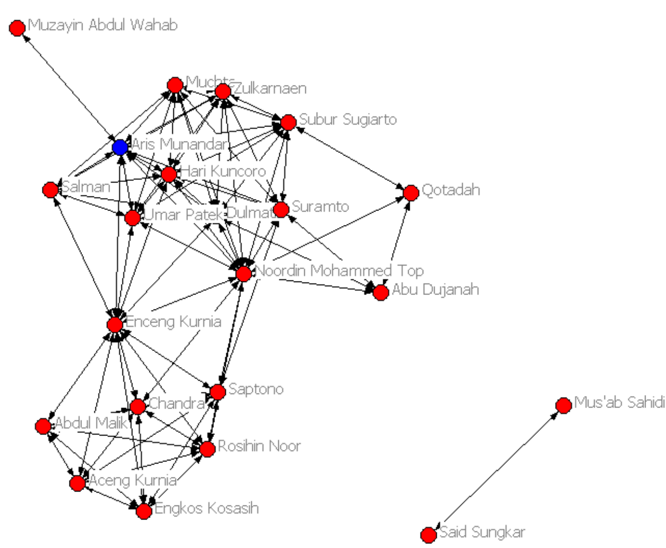


Figure 8.14: Noordin's Alive and Free Network with Cutpoints Highlighted

It does identify one cutpoint (Aris Munandar), but as you can see, Munandar simply connects the largest bi-component to a single individual (Muzayin Abdul Wahab). Thus, with regards to the Noordin network, bi-component analysis appears to be relatively unhelpful in identifying individuals whose removal would

disrupt the flow of information and other types of resources through the network. It does tell us, however, that both the entire network and the Alive and Free network are relatively immune to targeted attacks, suggesting that such an approach may not be the best strategy.

Bridges, Bi-components and Brokers in Pajek[42]

Net>Components >Bi-Components

In Pajek, we use the *Bi-Components* command found under the *Net> Components* submenu to detect a network's bi-components, bridges, and cutpoints. Upon selecting this command, a dialog box prompts us to specify the minimum size of the bi-components to be identified. While the default value three will identify the bi-components within the network, it will only report cutpoints that connect two or more bi-components of size three. If you select a minimum size of two, it will identify all bi-components, bridges, and cutpoints, including cut-vertices connecting bridges.[43] Pajek will ask if you want to overwrite the current network with a partition of lines. Select no, click OK, and the *Bi-Components* command generates a new network, two partitions and a hierarchy. If you draw the new network and in the Draw screen you select the *Options>Colors >Edges>Relation Number* command, the lines will be different colors, reflecting the bi-component of which they are a part. The first partition ("Vertices belonging to exactly one bi-component") indicates the number of the bi-component to which an actor belongs. Actors that do not belong to a bi-component (e.g., isolates) are assigned to class 0, and actors belonging to two or more bi-components (i.e., cutpoints) are assigned to class number 9999998. The second partition (Articulation points) indicates the number of bi-components to which a vertex belongs: 0 for isolates, 1 for actors that belong to exactly one bi-component, 2 for actors that belong to two bi-components and so on. Finally, the hierarchy shows the bi-components to which each actor belongs. Pajek uses hierarchy objects to store the bi-components because cutpoints can belong to two or more bi-components.

File>Hierarchy>Edit

Since bridges are components of size two in an undirected network without multiple lines, it is easy to find the bridges in the hierarchy of bi-components: open the Edit screen with the hierarchy of bi-components with the command *File>Hierarchy>Edit* or with the Edit button on the left of the hierarchy

[42] Portions of this section exception are adapted from W. de Nooy, A. Mrvar, and V. Batagelj. 2005. (Chapter 7) *Exploratory Social Network Analysis with Pajek*. Cambridge, UK: Cambridge University Press.
[43] Pajek's *Bi-Components* command treats directed networks as if they were undirected; in other words, in directed networks it identifies weak, rather than strong, components without cutpoints.

dropdown list. Next, click on the "+" sign to the left of the word "Root." This should produce a figure *similar* to Figure 8.15; this lists the six bridges and bi-components in the communication network among striking employees. The size of each subnetwork is reported between brackets, so to find the two bridges in the example double-click on the two subnetworks of size two in order to see the actors on either side of the bridge.

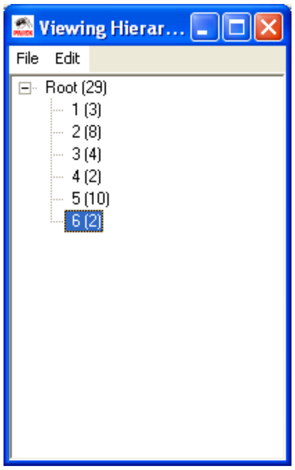


Figure 8.15: Pajek Hierarchy of Bi-components

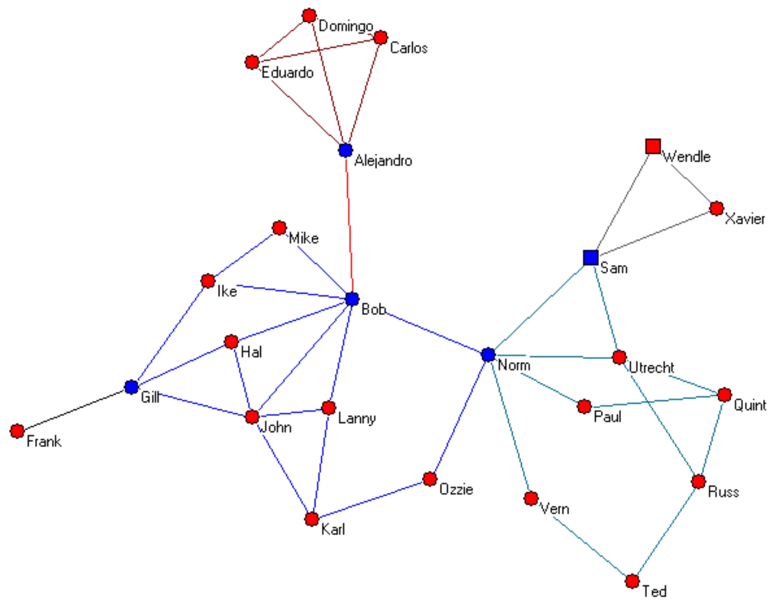


Figure 8.16: Strike Network with Cutpoints Highlighted

Draw>Draw-Partition

Next, open the draw screen using the *Draw>Draw-Partition* option from the main menu, making sure that the partition "Articulation points…" is highlighted in

---

Thus, if you symmetrize a directed network before executing Pajek's *Bi-Components* command, you will obtain exactly the same results.

the partition dropdown list. Now, you should see a drawing (Figure 8.16 above) where most of the employees are one color (e.g., red) whereas a handful of employees are another (e.g., blue). The blue actors are the cutpoints (i.e., articulation points).

A bi-component analysis of both the Noordin Combined network and the Noordin Alive and Free network yields identical results to those we found using UCINET and NetDraw. The Noordin Combined Network constitutes a single component without a cutpoint (so it is also a bi-component) while the Alive and Free network identifies three components (two of which are of size two) and one cutpoint. Figure 8.17 presents the Noordin Alive and Free network. As we saw above, at least in this case bi-component analysis is of little help in identifying actors whose removal would disrupt the flow of resources through the network. It does tell us, however, that the Alive and Free network appears to be relatively immune to a targeted attack, which suggests that such an approach may not yield the best results.

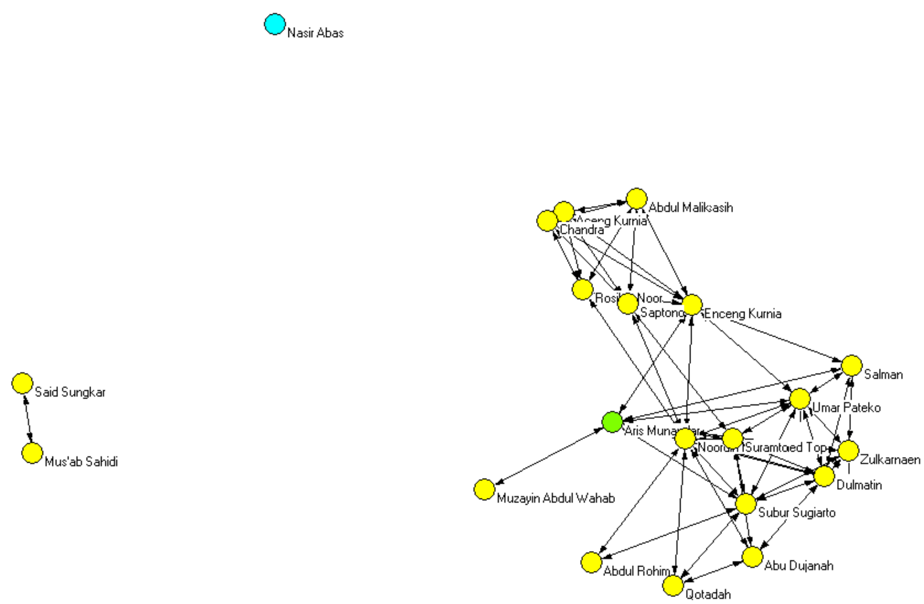


Figure 8.17: Noordin Alive and Free Network with Cutpoint Highlighted

8.3 Affiliations and Brokerage

Group affiliation is often an important factor in brokerage processes. For example, in brokering deals in Congress, U.S. Senators not only take into account their own interests and desires but also the political party of which they are apart. While they very much might want to support a particular legislative bill, their party membership may constrain what they are able to do and say. Roger Gould

and Roberto Gonzalez (Fernandez and Gould 1994; Gould and Fernandez 1989) have attempted to capture this dynamic by identifying five different types of brokerage roles that actors can play based their group affiliation ties: (1) coordinator, (2) itinerant broker/consultant, (3) representative, (4) gatekeeper and (5) liaison (see Figure 8.18 below – node color indicates group affiliation).

- *Coordinator* – Mediation between members of one group where the mediator is also a member of the group
- *Itinerant Broker/Consultant* – Mediator between members of one group where the mediator is not a member of the group
- *Representative* – Mediation between two groups where mediator regulates the flow of information or goods from his or her group
- *Gatekeeper* – Mediation between two groups where mediator regulates the flow of information or goods to his or her group
- *Liaison* – Mediation between two groups where mediator does not belong to either group

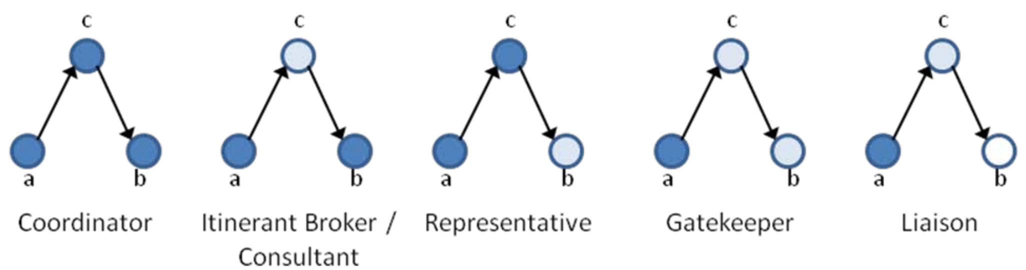


Figure 8.18: Brokerage Roles of Actor "c"

While when Gould and Fernandez originally conceived of these five types of brokerage roles in terms of directed networks, tie direction only distinguishes between representative and the gatekeeper brokerage roles. Thus, we can apply their brokerage roles algorithm to undirected networks, aware that each actor identified as a representative will also be a gatekeeper and vice versa.

Affiliations and Brokerage in UCINET and NetDraw

[UCINET] Network>Ego Networks >Brokerage roles

In UCINET we identify brokerage roles using its *Network>Ego Networks> Brokerage roles* command, which calls up the brokerage dialog box (Figure 8.19 below). As you can see because this routine needs to know the groups to which actors belong, not only do we need to indicate the network we want to analyze but also a partition file that indicates group membership. This partition can be based on predefined groups (e.g., Republican, Democrat) or on groups detected through one or more of the routines we discussed in the previous chapter. Here, I have chosen the Strike_groups partition to identify group affiliation for the Strike

network. Keeping UCINET's defaults and selecting OK generates an output log and saves its calculations in two attribute files (brokerage and relativebrokerage). UCINET's output log (Figure 8.20 below) is extensive. It first lists raw brokerage role scores (i.e., the count of how often each actor "plays" a particular role within the network). Next comes a series of block matrices for each actor, where each row/column refers to the various groups in the network (in the case of the strike network, three groups) and the numbers in the cells indicate how often each actor plays a brokerage role either between two groups or within a single group. Finally, it provides normalize brokerage scores (i.e., raw brokerage role counts are divided by expected counts – given a random network – based on network size.

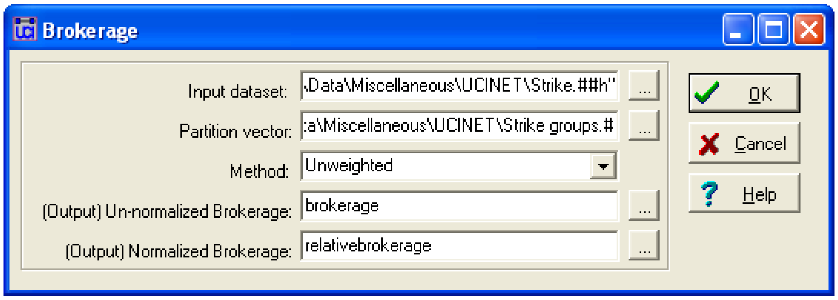
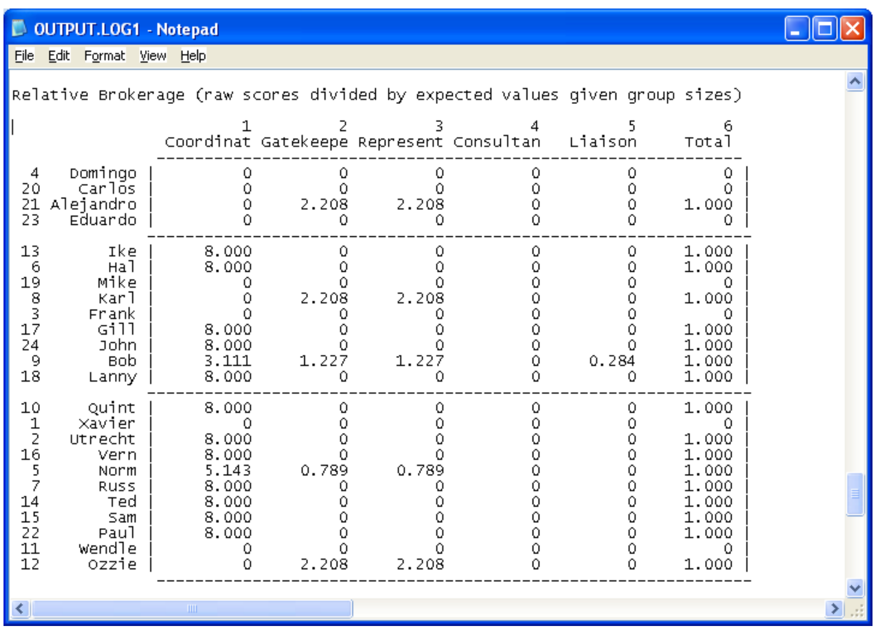


Figure 8.19: UCINET Brokerage Role Dialog Box



Relative Brokerage (raw scores divided by expected values given group sizes)

		1 Coordinat	2 Gatekeepe	3 Represent	4 Consultan	5 Liaison	6 Total
4	Domingo	0	0	0	0	0	0
20	Carlos	0	0	0	0	0	0
21	Alejandro	0	2.208	2.208	0	0	1.000
23	Eduardo	0	0	0	0	0	0
13	Ike	8.000	0	0	0	0	1.000
6	Hal	8.000	0	0	0	0	1.000
19	Mike	0	0	0	0	0	0
8	Karl	0	2.208	2.208	0	0	1.000
3	Frank	0	0	0	0	0	0
17	Gill	8.000	0	0	0	0	1.000
24	John	8.000	0	0	0	0	1.000
9	Bob	3.111	1.227	1.227	0	0.284	1.000
18	Lanny	8.000	0	0	0	0	1.000
10	Quint	8.000	0	0	0	0	1.000
1	Xavier	0	0	0	0	0	0
2	Utrecht	8.000	0	0	0	0	1.000
16	Vern	8.000	0	0	0	0	1.000
5	Norm	5.143	0.789	0.789	0	0	1.000
7	Russ	8.000	0	0	0	0	1.000
14	Ted	8.000	0	0	0	0	1.000
15	Sam	8.000	0	0	0	0	1.000
22	Paul	8.000	0	0	0	0	1.000
11	Wendle	0	0	0	0	0	0
12	Ozzie	0	2.208	2.208	0	0	1.000

Figure 8.20: UCINET Brokerage Role Output Log

Figure 8.20 reproduces a relative brokerage portion of UCINET's output log for the strike network. As you can see several of the employees fill coordinator roles within the network. However, a coordinator is one who mediates between

members of one group where the mediator is also a member of the group, so in terms of brokering between groups, this role is relatively unhelpful.

By contrast, five actors (Alejandro, Karl, Bob, Norm and Ozzie) fill the gatekeeper and representative roles (which are the same in undirected networks), which are indicators of mediation roles between groups. These findings are consistent with what we discovered earlier with our structural holes and bi-component analyses. However, Bob also fills coordinator and liaison roles, which suggests that in terms of brokerage, he may be the most important actor in the network.

[NetDraw]
File>Open>Ucinet dataset >Network

File>Open>Ucinet dataset >Attribute data

Properties>Nodes>Symbols >Size>Attribute-based

Currently, NetDraw does not include a brokerage role routine. Nevertheless, we can visualize each of the five different brokerage role schemes in NetDraw by opening both the strike network file (first) and the "relativebrokerage" partition file (second), and then varying the seize of the network nodes by each of the five brokerage roles. Figure 8.21 indicates the relative brokerage potential in terms of the liaison brokerage role. Indeed, relatively high scores in terms of liaison indicate that an actor possesses brokerage potential between two groups when the actor is not a member of either group. This suggests that Bob plays a key role in mediating between the older English-speaking group and the Spanish-speaking group.

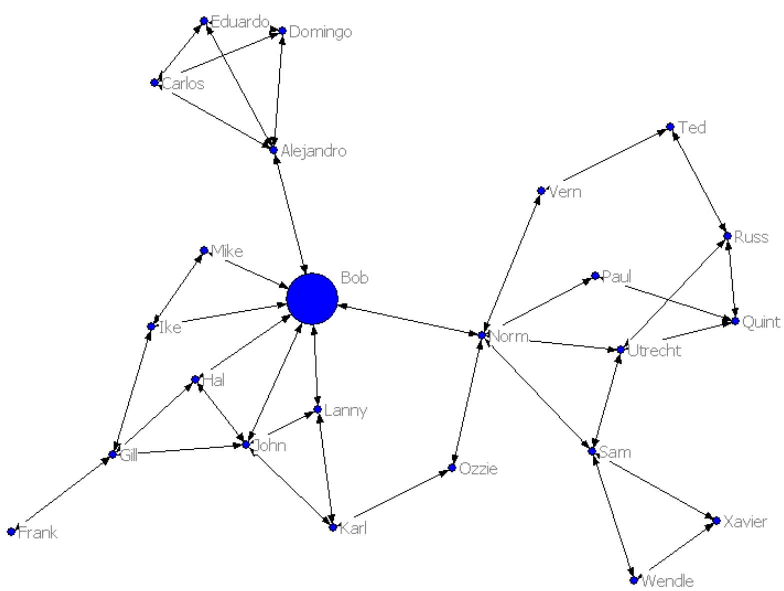


Figure 8.21: Liaison Brokerage Potential of Strike Network

Let us now examine the brokerage potential of the Combined Noordin Network using the factional analysis results obtained in the previous chapter. The output is voluminous, which makes it difficult to interpret at a glance. It does

indicate that only a handful of individuals fulfill coordinator and itinerant/ consultant roles; we will ignore these for now because let us assume (for the sake of brevity) that we are only interested in identifying actors who fulfill brokerage roles between groups. A visual analysis of the three different types of roles suggest that the most variation exists in terms of the liaison role as illustrated by Figure 8.22 below, where size of node indicates liaison brokerage potential and color of node indicates that factions to which each actor belongs. As we did in the previous chapter, we used principal components analysis to visualize the network.

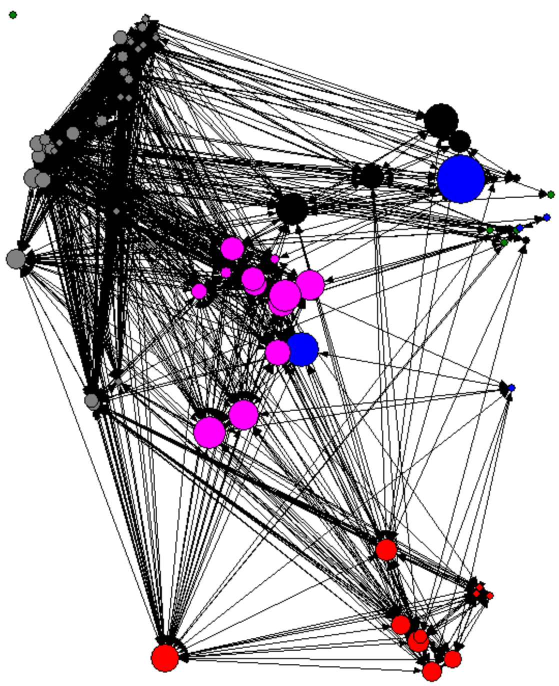


Figure 8.22: Liaison Brokerage Potential of Combined Noordin Network

Although two of the actors (red node, lower left and blue node, upper right) appear to harbor some brokerage potential, there a number of other network members who appear to be in a brokerage position. At least in this case, then, visualization does not appear to help us much in identifying key actors in terms of brokerage. Thus, it seems incumbent to turn our attention to the ranking of actors by the various brokerage metrics. That is the purpose of Table 8.1 (next page), which presents the rankings of the top 10 "brokers" in the Combined Noordin Network. Here raw brokerage scores (rather than relative brokerage scores) were used and the actors are ranked in terms of the sum of the scores. Because the gatekeeper and representative roles are identical when working with undirected data, those scores were only counted once in computing the ranking. Not unexpectedly, Noordin Top ranks in the top ten, but he does not score the highest in terms of overall brokerage potential. Iwan Dharmawan does. Interestingly, Dharmawan

ranks sixth in terms of degree centrality, which is further evidence that one metric does not tell all.

Name	Coordinator	Itinerant Broker Consultant	Gatekeeper Representa-tive	Liaison	Total
Iwan Dharmawan	0	269	18	756	1,043
Noordin Top	0	598	12	378	988
Azhari Husin	0	431	0	188	619
Jabir	0	422	2	184	608
Abdullah Sungkar	0	458	2	92	552
Ahmad Ridho	0	399	6	146	551
Abu Bakar Ba'asyir	0	454	0	96	550
Usman bin Sef	0	332	0	108	440
Heri Golun	0	126	0	288	414
Mohamed Rais	0	320	2	68	390

Table 8.1: Brokerage Potential Ranking of Noordin’s Network

Affiliations and Brokerage in Pajek

[Pajek] >Operations >Brokerage Roles

In Pajek the *Operations>Brokerage Roles* command identifies brokerage roles. Issuing this command generates five new partitions, one for each type of brokerage role, all of which are added to the Partition dropdown list. The class number of an actor in a partition specifies the number of times this actor plays the corresponding brokerage role. We can call up a frequency table for each partition using the *Info>Partition* command, and we can check the individual scores of each actor using Pajek's *File>Partition>Edit* command. To visualize the network where node size reflects the number of one of the brokerage roles, we need to first the brokerage partition into a vector. Above when were exploring NetDraw's capabilities, we visualized the liaison brokerage roles, so here let us examine the gatekeeper roles. With the "Gatekeepers in N1 according to C1 (24)" partition showing in the partition dropdown list, we first select the *Partition>Make-Vector* command. Next, with the strike network data showing in the Network dropdown list, the "Strike_groups" partition highlighted in the Partition dropdown list, and the gatekeeper vector we just created displayed in the Vector dropdown list, we visualized the network using Pajek's *Draw>Draw-Partition-Vector* command, which produced the following network map (Figure 8.23).

Info>Partition

File>Partition>Edit

Partition>Make Vector

Draw >Draw-Partition-Vector

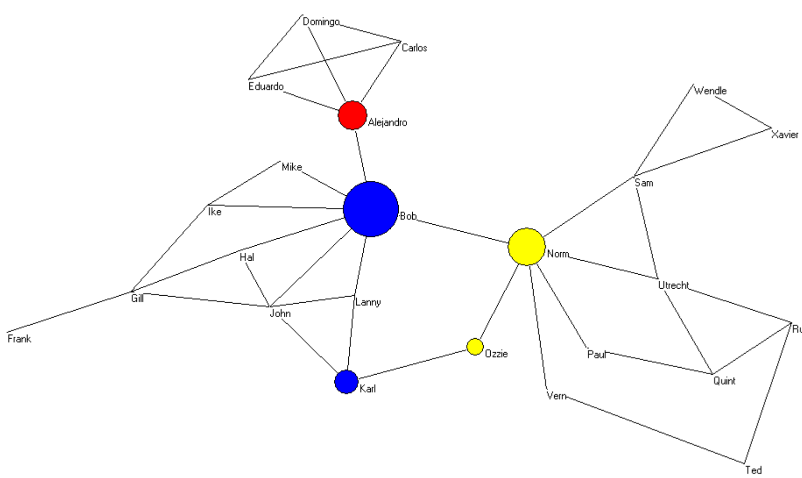


Figure 8.23: Pajek Drawing of Strike Network with Node-size Relative to Gatekeeper Brokerage Role Potential

As indicated in the output displayed in Figure 8.20, Alejandro, Karl, Bob, Norm and Ozzie fill gatekeeper roles in the strike network, a role that indicates an actor is in the position to mediate between groups. Once again, however, Bob appears to be the most important player in the Strike network, so it is not terribly

surprising to learn that after the consultant finished his analysis, management first approached Bob in their attempts to resolve the issues surrounding the strike.

Operations>Extract >Partition

Since we examined the Combined Noordin Network with UCINET and NetDraw, let us now turn our attention to Noordin's Alive and Free network. For an affiliation partition, we can use the K-Core analysis partition created earlier, except that we will only analyze brokerage potential among those members who were members of the 6 and 8-core groups. Thus, we need to first extract those cores from the larger network using Pajek's *Operations>Extract> Partition* command. In the dialog box that this command calls up, we tell Pajek that we want clusters 6 and above by entering "6-*" (see Figure 8.24 below). Clicking OK generates a new network and a new partition consisting of seventeen actors.

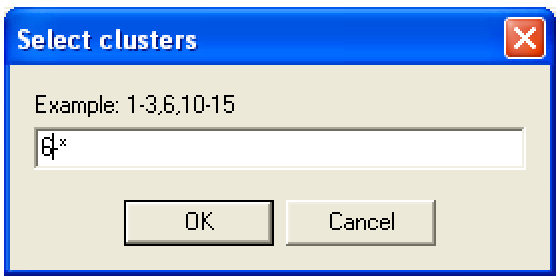


Figure 8.23: Pajek's Extract Based on Partition Dialog Box

Operations>Brokerage Roles

Info>Partition

File>Partition>Edit

Partition>Make Vector

Draw >Draw-Partition-Vector

Next, we need to calculate the number of brokerage roles using Pajek's *Operations>Brokerage Roles* command. As before Pajek creates five new partitions, one for each type of brokerage role. An examination of the various partitions using both the *Info>Partition* and *File>Partition>Edit* commands indicates that only the coordinator and gatekeeper/representative roles are present in the network. Only one actor (Enceng Kurnia) fills the coordinator role, while several fill the gatekeeper role. Thus, we will visualize the Alive and Free network varying the size of nodes by their gatekeeper brokerage potential and the color of nodes based on the k-core group to which they belong. To do this we need to first convert the gatekeeper partition to a vector just as we did above when working with the strike network (i.e., using the *Partition>Make Vector* command). Next, after insuring that the seventeen-node network (i.e., the network we just extracted) and partition as well as the gatekeeper *vector* are highlighted in their respective dropdown boxes, we visualize the network using Pajek's *Draw>Draw-Partition-Vector* command, which results in a drawing similar to Figure 8.24 (next page). As we can see Enceng Kurnia and Noordin Top appear to play pivotal roles in the brokerage of resources between the two groups, suggesting that their removal from the network could seriously disrupt the ability of the network to mobilize.

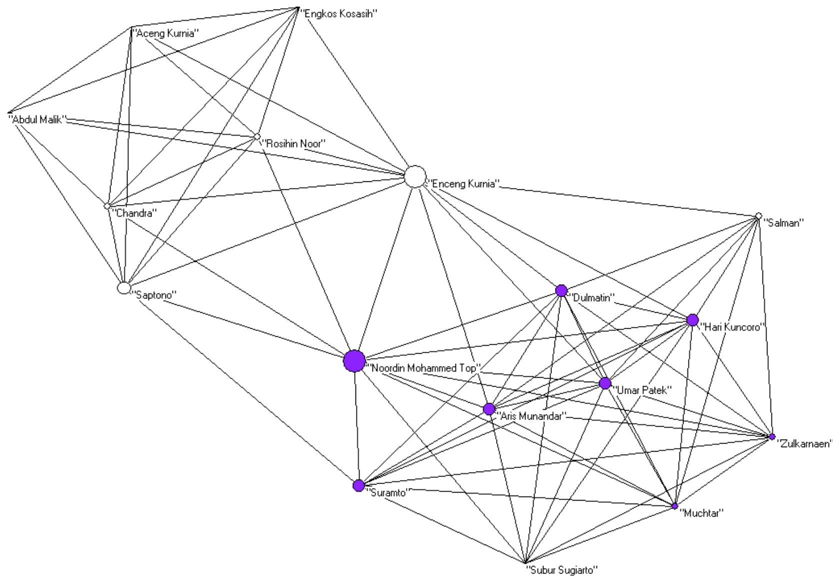


Figure 8.24: Gatekeeper Roles for Noordin's Alive and Free Network

8.4 Conclusion

In this chapter we have examined three broad ways of identifying the brokerage potential of actors within a network. We began by looking at Burt's (1992a; 1992b) notion of structural holes, which we saw is based on Mark Granovetter's (1973; 1974) work on weak ties. Burt argues that when it comes to identifying brokerage potential it is not the type of tie that is important but rather the gaps in the social structure. Next we examined how we can use a technique known as bi-component analysis to identify the actors and bridges within a network whose removal would disconnect the network. Finally we explored an algorithm that assumes that brokerage is a function of the different groups with which actors are affiliated; thus, not only does it require network data, it also requires attribute data indicating the specific groups to which actors belong. As we saw, these algorithms do not always detect actors who have the potential for brokering the flow of resources through a network. Sometimes one will work while another does not and vice versa. What this illustrates (once again) is that social network analysts cannot rely on a single "magic bullet" algorithm. Instead, we need to explore (both light and dark) networks using a variety of algorithms; otherwise we may never get at the underlying social structure of the network we are attempting to track and disrupt.

CHAPTER 9

STRATEGIC CHOICES IN DISRUPTING DARK NETWORKS

It is hard to argue with the observation that in recent years social network analysis has enhanced our understanding of how dark networks are structured and offered potential strategies for their disruption. We have learned, for example, that the September 11, 2001 terrorist network was a relatively sparse network in that its members had relatively few ties to others in the group (Krebs 2001). Apparently, some did not even know some of the others who were on the same flight as themselves. In spite of the network's lack of connectedness, however, a handful of those involved (e.g., Mohamed Atta and Nawaf Alhazmi) possessed key ties to others that allowed them to control, broker and facilitate the flow of information and other resources through and across the network (Krebs 2001). The terrorist network that carried out the March 11, 2004, Madrid train bombings displayed similar dynamics (Rodriguez 2005) It appears that it was characterized by loose or weak ties (Granovetter 1973) that enabled its various cells to maintain operative ties with the larger network while remaining relatively isolated from and unknown to one another. This probably helped the network remain relatively invisible to counter terrorism efforts and provided it with a degree of stability if (and when) group members were captured since most possessed little or no knowledge of the network's overall structure.[44]

Network analysis of dark networks took a giant leap forward with Marc Sageman's (2004a; 2004b) analysis of the global salafi jihad (GSJ).[45] Not only did it challenge stereotypes that many hold regarding terrorists,[46] but it found that the GSJ exhibits network dynamics that researchers have discovered about other groups (religious and otherwise). For example, like most social movements (Lofland and Stark 1965; McAdam 1986; Snow, Zurcher and Ekland-Olson 1980; Stark and Bainbridge 1980), the GSJ recruits primarily through social ties, in

---

[44] For similar reasons suicide bombers appear to be kept on the periphery of most terrorist networks (Pedahzur and Perliger 2006); since they are the most likely members to be caught, it is important for the network's survival that they possess as little information about the group as possible.

[45] By global salafi jihad Sageman means those Muslims who believe that in order for Islam to recapture the economic, cultural and military preeminence that it once enjoyed, not only do Muslims need to return to the practices of their devout ancestors (*salaf* in Arabic), but that it is permissible to use violence against both the near enemy (Muslim states that have fallen away from the true faith) and the far enemy (the West, in particular the U.S and Israel). When speaking of the global salafi jihad, he generally is referring to terrorists who focus their efforts on the West.

[46] For example, contrary to the conventional wisdom, most terrorists tend to be well-educated and earned their degrees at secular, not religious, institutions.

particular, through kinship and friendship ties (Sageman 2004b).[47] The GSJ also displays the characteristics of scale-free networks (Barabasi 2002; Barabasi and Bonabeau 2003) where most actors have very few ties, while a handful ("hubs") have very many.[48] Research into scale-free networks has discovered that they are relatively immune to random failures but are quite vulnerable to targeted attacks. This is because when a random actor fails in a scale-free network, little damage is inflicted on overall structure of the network because the actor probably has very few ties to other actors in the network. However, because hubs have so many ties to other actors, the simultaneous failure of 10 to 15% hubs can crash (i.e., disconnect) the network (Barabasi and Bonabeau 2003:56). This led Sageman to argue that the United States should focus its efforts on taking out hubs rather than randomly stopping terrorists at our borders. "[The latter] may stop terrorists from coming here, but will leave the network undisturbed. However… if the hubs are destroyed, the system breaks down into isolated nodes. The jihad will be incapable of mounting sophisticated large scale operations like the 9/11 attacks and be reduced to small attacks by singletons" (Sageman 2004a). This is easier said than done, however. Terrorist networks appear to be remarkably resilient and self-healing (Tsvetovat and Carley 2005), and research suggests that when a highly central actor is eliminated, it is usually quickly replaced by another, highly central and/or similar actor (Pedahzur and Perliger 2006; Tsvetovat and Carley 2005). This has led social network analysts to explore more complex methods for destabilizing terrorist networks (Carley, Lee and Krackhardt 2002; Carley, Reminga and Kamneva 2003).

To date research of terrorist networks using SNA has tended to focus on individual level social networks, in particular, key actors within the network who score high in terms of centrality or whose structural location (i.e., their location within the overall network) allows them to broker information and/or resources within the network. However, while focusing on key individuals may be intuitively appealing and might provide short-term satisfaction, such a focus may at times be misplaced and, in fact, could make tracking, disrupting and destabilizing dark networks more difficult than it already is. As Ori Brafman and Rod Beckstrom (2006) have pointed out, targeting key players in decentralized organizations seldom shuts down the organization. Instead, it only drives them

---

[47] Moreover, if previous research is any guide (Stark and Bainbridge 1980), members recruited through these ties are also the least likely ones to defect from the group.

[48] Barabasi and his colleagues (Barabasi 2002; Barabasi and Bonabeau 2003) coined the term, "scale-free," because scale–free networks do not follow a typical bell curve distribution (in terms of ties per actor) where the mean number of ties per actor in the network provides a scale that provides useful information about the network. In scale–free networks, the mean number of ties per node is essentially meaningless; hence, the network is scale–free.

further underground, causing them to become even more decentralized, which in turn makes them even harder to target.[49] For example, attempts at combating the peer-to-peer (P2P) music industry may have successfully shut down Napster, they did not eliminate the P2P music industry. Instead it has simply become more decentralized as new players have joined the industry that cannot be shut down because they simply cannot be found (Brafman and Beckstrom 2006:11-27). Take the Animal Liberation Front (ALF) as another example. Like most decentralized organizations, they have been relatively immune to targeted attacks. The FBI has gone after it but has met with little or no success. Its efforts have only driven the ALF more underground and made them even more decentralized (Brafman and Beckstrom 2006:135-143). Put simply, then, targeting key players within a dark network may not yield the results many expect.

However, if targeting key players within a dark network is not always the best approach, then what policies and strategies should we adopt? The answer to that question is not as straightforward as one would hope although it should involve exploratory approach that takes into consideration the overall topography of the network (e.g., density, clustering levels), different points of entry into the network (e.g., individual, subgroup, and group levels), and different approaches (e.g., direct action, information operations, psyop operations). Here we will briefly consider how each of these factors should play into our strategic decisions.

9.1 Network Topography

In the fifth chapter we saw how research building on Mark Granovetter's (1973; 1974) work on the strength of weak ties suggests that actors' networks range from networks consisting primarily of strong, redundant ties (i.e., provincial networks) to networks consisting primarily of weak ties (i.e., cosmopolitan networks). Moreover we encountered evidence that suggests that the most productive or efficient networks are those that maintain a balance of weak and strong ties. Put differently, in order to be successful, networks that are either too provincial or too cosmopolitan are not as successful those that lie somewhere between the two extremes. This suggests that we may want to adopt strategies

[49] In fact, it may exacerbate what Marc Sageman (2008) has come to call the "leaderless jihad," namely the numerous independent and local groups that have branded themselves with the Al Qaeda name and are attempting to emulate bin Laden and his followers by conceiving and executing terrorist operations from the bottom up. Sageman argues that Al Qaeda Central is down to a few dozen individuals who are on the run in northwest Pakistan. Instead of recruiting and training terrorists and authorizing and coordinating terrorist attacks, it now serves primarily as a source of inspiration.

that "push" dark networks toward the tails of the provincial-cosmopolitan continuum (Figure 9.1).

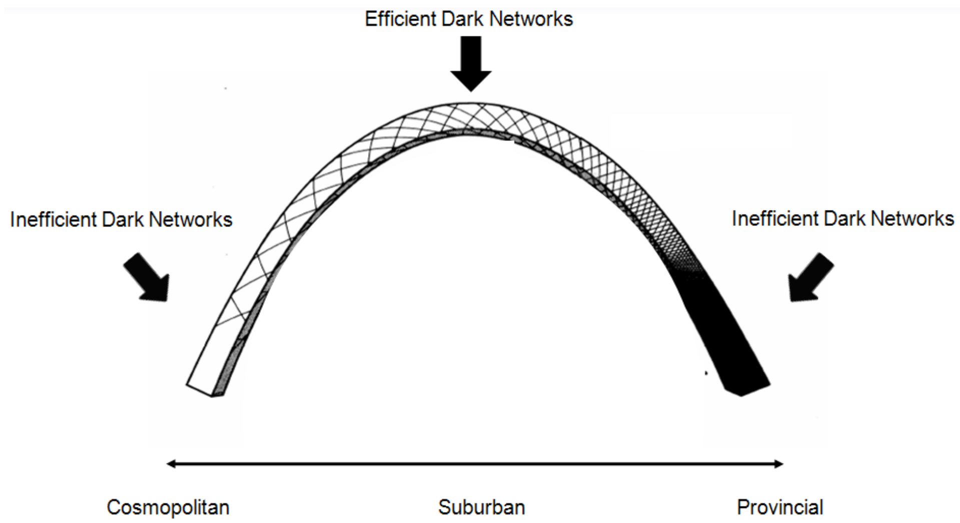


Figure 9.1: Relationship Between Network Topography and Network Efficiency

This means that in some cases targeting key players may prove to be a beneficial strategy; that is, if we are encountering a network that already leans to the cosmopolitan side of the continuum, identifying key players for elimination or capture may force the network to become even more cosmopolitan, making it difficult for it to mobilize individuals and resources. While Brafman and Beckstrom (2006) may be correct that targeting key players in decentralized networks makes them even more difficult to track and disrupt, it is also true that networks with few connections will find it difficult to mobilize with any great effectiveness. As Brian Jackson of the RAND Corporation has pointed out, if by targeting key players we reduce Al Qaeda to setting fires like the Animal Liberation Front, then we should probably consider our disrupting efforts a success.[50] That said, we probably should not target key players in a network that leans to the provincial side of the continuum. Doing so could push the network to adopt a network structure that is more effective than it already is. Instead, when facing a relatively provincial network, we should probably adopt strategies that force the network to become even more provincial.

Thus, to return to a point made at the conclusion of Chapter 5, an important thing to keep in mind when developing strategies for disrupting dark networks is that no two networks are exactly alike, and their overall characteristics will most likely have an impact on their performance and efficiency. Thus, their overall

[50] Personal communication, September 20, 2008.

structure should provide us with clues as to what strategies we will want to choose for rendering them less effective.

9.2 Multiple Points (and Levels) of Entry

We have already seen that focusing on key players is not always the best approach, which suggests that we will want to consider other points of entry into the network. For example, we may want to employ amnesty and reconciliation strategies[51] that attempt to reintegrate peripheral players back into the wider community (and out of the dark network) since they are less committed and more likely to leave a group than core members (Popielarz and McPherson 1995; Stark and Bainbridge 1980). Such an approach could weaken a network from the outside in, similar to the peeling of an onion. We can use a number of metrics to identify a network's peripheral players. Actors who score low in terms of the various centrality scores that we examined in Chapter 6, actors who are not part of a network's core cohesive subgroup (see Figure 7.26 above), and/or actors who score low in terms of brokerage potential (e.g., using Burt's structural holes measure) could be considered peripheral players within the larger network.

Another consideration is to move beyond the individual level and consider the networks of organizations, institutions and groups that generate and sustain terrorist networks. As social movement theorists have repeatedly pointed out, insurgencies do not arise from and are not sustained by unorganized groups or isolated individuals (McAdam 1982, 1999; McAdam, McCarthy and Zald 1988, 1996; Smith 1991:60). Instead, formal and informal networks of groups and organizations not only shape (and sustain) the moral outrage that drives people to join insurgencies in the first place, but they also facilitate the coordination, mobilization and deployment of insurgent activities (Smith 1996). Curiously, though, social network analyses of dark networks have paid little or no attention to interorganizational networks even though they play such a central role in their emergence and mobilization. Add to this the fact that disagreements among and fighting between insurgent organizations are a primary reason why insurgencies often fail (McAdam 1982; 1999), one would think that those of us who seek strategies to destabilize and disrupt dark networks would broaden our focus to include interorganizational networks as well.

Like at the individual level, at the organizational level we can focus on key and/or peripheral actors. Figures 9.2 and 9.3 present the Noordin institutional

---

[51] As an example, take the case of Nasir Abas, a former member of Noordin's network who now regularly meets with captured terrorists and encourages them to leave their violent pasts behind. He has apparently met with some success (Mydans 2008).

network, with central actors highlighted (i.e., circled) in the former and peripheral actors highlighted in the latter.[52]

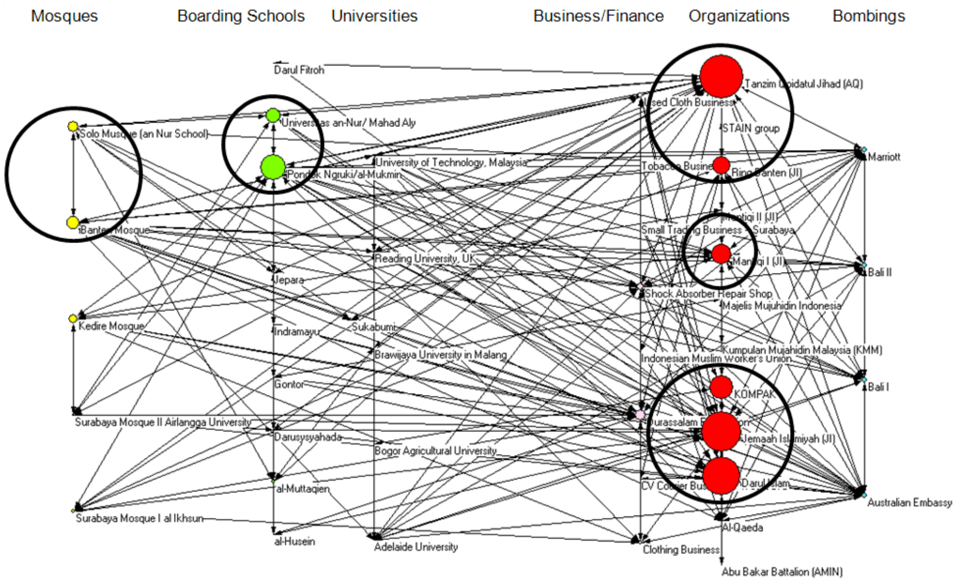


Figure 9.2: Noordin Institutional Network (Central Actors Highlighted)

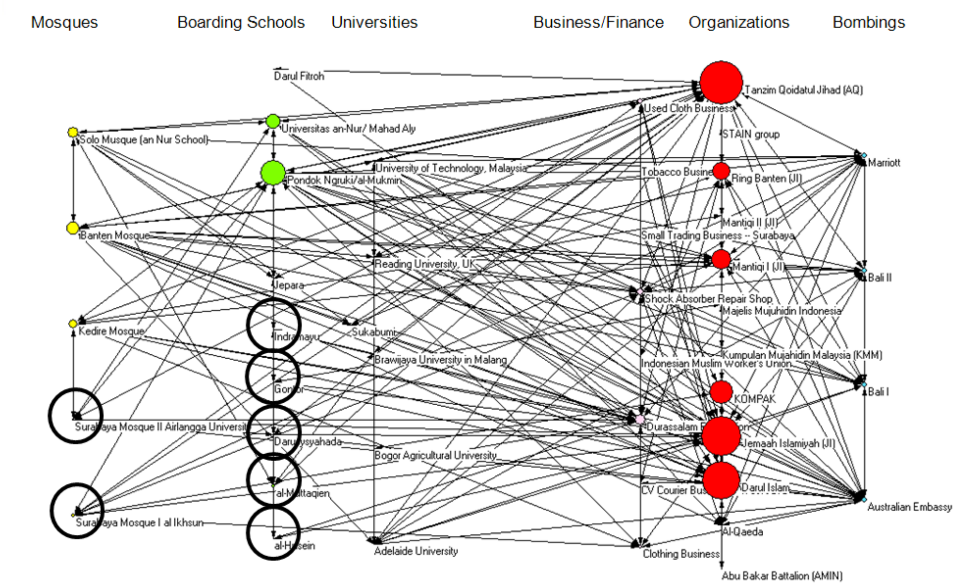


Figure 9.3: Noordin Institutional Network (Peripheral Actors Highlighted)

On the far left are what social movement theorists refer to as “feeder” organizations and “movement midwives” (in this case, mosques, boarding schools

[52] Node size varies in terms of betweenness centrality

and universities), which are those organizations that help to foster an insurgency's initial emergence and provide an ongoing supply of members (Smith 1996). Just to the right of center are that organizations that are instrumental in facilitating the mobilization and deployment of Noordin's terrorist activities: the businesses and foundations the help facilitate the flow of resources (e.g., funds, equipment) to the network and the terrorist organizations help coordinate various activities. On the far right are simply the various operations in which members of Noordin's terrorist network have participated.

Clearly these organizations vary in terms of centrality. As with individual networks, we probably want to think twice about targeting for elimination the more central organizations. The benefits of such actions may prove beneficial only in the short term, ultimately leading the network to become increasingly decentralized. For example, once authorities learned that a number of terrorists came from the Luqmanul Hakeim school (not shown in Figures 9.2 and 9.3), they shut it down. However, there is no evidence to suggest that this decreased the flow of individuals to terrorist networks such as Noordin's. Thus, we may be better served by attempting to infiltrate the more central organizations in order to improve intelligence gathering and to possibly breed distrust between these organizations. Less central organizations may prove to be susceptible for reintegration back into the wider community (e.g., convincing local imams that preaching violent jihad is not the answer).

9.3 Strategic Options

Finally, let us briefly consider what strategic options may be available to strategists, regardless of what point of entry we may be targeting. Implicit in much of the work on identifying key players in a network is the use of direct or kinetic action in order to eliminate or capture such key players. This is certainly reflected in how Sageman (2004a; 2004b) draws on scale-free network research (Barabasi 2002; Barabasi and Bonabeau 2003) to argue that we should focus our efforts on eliminating hubs rather than randomly stopping terrorists at our borders. It is unclear, however, whether direct action is a winning approach when employed by strong actors (e.g., the United States) against weaker opponents (e.g., Al Qaeda) that use indirect strategies. Ivan Arreguin-Toft's (2005) analysis of over 200 historic cases suggests that since dark networks almost always use indirect strategies, in order to defeat them, so must we (Borer 2008).

Since in previous sections we have already briefly considered a few such indirect strategies, we will limit our analysis here to the examination of two more examples, taken from student projects from the "Tracking and Disrupting Dark

Networks" course taught at the Naval Postgraduate School in Monterey, California. In the first project (Defeyter and Abell 2008) the authors initially identified what they called Noordin's "Alive and Effective" network, which consisted essentially of all members in the network who were not only alive but also in a position to be effective members within the network (e.g., they were not on the run from the authorities). This analysis yielded two distinct cells. Based on this analysis, they then suggested a strategy where a member of one cell (Abu Fida), who sits in a position of brokerage (e.g., he sits aside a structural hole), is arrested and then released within 24 to 48 hours. Upon his release a member of the other cell (Chandra), who also sits in a position of brokerage, is arrested and a suspiciously large sum of money is placed into Abu Fida's accounts. At the same time, information is leaked by the authorities, which suggests that Fida has become an informant. This strategy is then repeated, except that the order of cells targeted is reversed; that is, a key player in Chandra's cell is arrested (Saptano) and released upon which a large sum of money is placed in Saptano's account and a key player from Fida's cell (Suramto) is arrested. The goal of this strategy is to breed distrust among members of Noordin's network, so that its members turn in on themselves rather than directing their efforts (and anger) at innocent civilians.

In the second example the authors (Pedersen and Gimmingsrud 2008) actually considered three alternative strategies, weighing the advantages and disadvantages of each. Their first strategy examined whether Noordin's network is, in fact, a scale-free network, assuming that if it is, then targeting its hubs (i.e., key players) could disrupt the network. The second strategy also considered the possibility of targeting key players, except this time the authors defined "key" in terms of actors' critical knowledge (i.e., bomb makers). The final strategy considered by the authors focused on providing less committed members a way out of the network through a well-publicized amnesty program or an offer of reduced punishment for defectors (CTC 2006:43). They ultimately settled on the third strategy, noting that the removal of key actors based on centrality (first strategy) did not appear to cause any meaningful disruption of the network,[53] and while the removal of key actors based on critical knowledge might degrade the network in the short term, in the long run it was likely that Noordin would be able to recruit new bomb makers. By contrast their analysis of what would happen to the network if less committed members chose to leave the network suggested that the network's long-term effectiveness would be degraded.

[53] Using various measures of centrality (degree, closeness, betweenness, eigenvector), the authors hypothetically removed the top 12 central actors (15%) in Noordin's network and found that the network remained largely intact.

9.4 Conclusion

In this final chapter we have noted that to date the strategic use of social network analysis to track and disrupt dark networks has tended to focus on direct action targeting key individuals. As we have seen, while this approach may sometimes yield positive results, there is no guarantee that it will. Thus, we need to also consider multiple strategies that take into account the overall topography of the network, different points of entry into the network, and different approaches (e.g., direct and indirect). What should be clear is that social network analysis does not provide a single “magic bullet” algorithm that will solve all our problems. Instead, using social network analysis to track and disrupt dark networks takes time and patience. Nevertheless, when used correctly, it will undoubtedly provide possible solutions to complex problems.

APPENDIX 1

NOORDIN TOP'S NETWORK DATA

These data were drawn from "Terrorism in Indonesia: Noordin's Networks," a publication of the International Crisis Group (Asia Report #114, 5 May 2006). The data were structured and analyzed by Defense Analysis students in the course "Tracking and Disrupting Dark Networks" under the direction of Professor Nancy Roberts, Co-Director of the CORE Lab, and updated by Dr. Sean Everton.

1. TERRORIST/INSURGENT ORGANIZATIONS (LACEY/SEARS) – Orgs.##h

Definition Terrorist/Insurgent Organization:

A terrorist/insurgent organization is defined as an administrative and functional system, whose primary common goal is the operational conduct of terrorist/ insurgent activities, consisting of willingly affiliated claimant members. Factions and offshoots will be considered separate from their parent organization.

List of Terrorist/Insurgent Organizations:

1. Abu Bakar Battalion (AMIN)
2. Al-Qaeda (AQ)
3. Darul Islam (DI)
4. Islamic Defenders Front (FPI)
5. Jemaah Islamiyah (JI)
6. KOMPAK
7. Kumpulan Mujahidin Malaysia (KMM)
8. Majelis Mujahidin Indonesia
9. Mantiqi I (JI)
10. Mantiqi II (JI)
11. Mantiqi III (JI)
12. Mujahidin Kayamanya
13. Ring Banten (DI)
14. Tanzim Qoidatul Jihad (AQ)
15. STAIN Group

Two-mode network 79 X 15

2. EDUCATIONAL RELATIONS (SHANNON/GLENN) – Schools.##h

Definition of Educational Relations:

Educational relations are defined as schools where individuals received formal education.

List of Schools:

1. Adelaide University
2. al-Husein – *pesantren* (Islamic boarding school) in Indramayu, West Java
3. al-Muttaqien
4. Sukabumi
5. Bogor Agricultural University
6. Brawijaya University in Malang
7. Darusysyahada – *pesantren* (Islamic boarding school) in Boyolali
8. Gontor
9. Indramayu
10. Jepara
11. Luqmanul Hakeim – *pesantren* (Islamic boarding school) in Johor, Malaysia
12. Pondok Ngruki/al-Mukmin – *pesantren* (Islamic boarding school) in Ngruki, Central Java
13. Reading University, UK
14. Universitas an-Nur/ Mahad Aly – *pesantren* (Islamic boarding school) in Solo
15. University of Technology, Malaysia

Two-mode network 79 X 15

3. COMMUNICATION RELATIONS (THOMAS/ACOSTA) – IntComs.##h

Definition of Internal Communication:

Internal communication is defined as the relaying of messages between individuals and/or groups inside the network through some sort of medium.

One-mode matrix, 79 x 79

4. MEDIUM FOR EXTERNAL COMMUNICATION: (THOMAS/ACOSTA) – ExtComs.##h)

Definition of Medium for External Communication:

Medium for external communication is defined as the various means utilized to relay information between the terrorists and those outside the network.

List of Mediums:

1. Computer-based messages—such as website (www.anshar.net), network cafes, and email.
2. Print media—such as the KOMPAK magazine
3. Codes and passwords—to send material from prison.
4. Videos—for bomb making and training to recruit outside members.
5. Undefined—any other, undefined, yet referenced type of communication between actors

Two-mode matrix, 79 x 5

5. KINSHIP RELATIONS (DAVE AND GLENN) – Kinships.##h

Definition of Kinship:

Kinship is defined as a family connection based on marriage. Kinship will include current marriages and past marriages due to divorces and/or deaths.

One-mode matrix, 79x79

6. TRAINING RELATIONS (MIRASLOV AND MIKE) – Training.##h

Definition of Training Relations:

Participation in any specifically designated activity that teaches the knowledge, skills, and competencies of terrorism. Training does not include participation in a terrorist sponsored act or mujahedeen activity in places such as Afghanistan, Bosnia, Chechnya, or Iraq unless the individuals' presence was to participate in a specifically designated training camp or base in one of these areas.

List of Training Locations:

1. Post-Bali Mil Refresh Training
2. Jan 04 Bomb Making
3. Jun 04 Bomb Making
4. 03 Rois Training
5. May 04 Training
6. 99 Mindanao Training

7. Australian Embassy Religious Training
8. 01-02 Ujunj Kulon Training
9. 03 Mindanao Training
10. Oct 99 Waimurat, Buru Training
11. Jul 04 West Ceram
12. Azhari Apprenticeship
13. Solo course
14. Training for Bali II in “Selera” restaurant

Two-mode matrix, 79x14

7. RECRUITING RELATIONS (MIRASLOV/MIKE) – Recruit.##h

Recruiting Relations Defined:

Contact between two individuals for the purpose of enlisting new members for terror-related activities. Recruiting relations are only relevant where the attempt to enlist members has been successful. Failed recruiting attempts are not included.

One-mode matrix, 79 x 79

8a. BUSINESS RELATIONS (ADRIANO/CARL) – Biz.##h

Definition of Business Relations:

Profit and non profit organizations that employ people.

Types of Businesses:

1. Shock Repair Shop-- the automobile shop that repaired shock absorbers
2. CV Courier Business—business that specializes in transfer of information and products
3. Indonesian Muslim Workers Union
4. Tobacco Business—firm that grows tobacco
5. Small Trading Business—exchange of goods
6. Used Cloth Business—the collection and sale of used cloth for industrial purposes
7. Clothing Business—making and selling clothing

Two-mode matrix, 79 x 7

8b. BUSINESS RELATIONS (ADRIANO/CARL/EVERTON) – Biz.##h

Definition of Business Relations:

Profit and non profit organizations that employ people.

Types of Business and or Finance Operation:

1. Clothing Business—making and selling clothing
2. CV Courier Business—business that specializes in transfer of information and products
3. Durassalam Foundation
4. Indonesian Muslim Workers Union
5. Shock Repair Shop-- the automobile shop that repaired shock absorbers.
6. Small Trading Business—exchange of goods
7. Tobacco Business—firm that grows tobacco
8. Used Cloth Business—the collection and sale of used cloth for industrial purposes

Two-mode matrix, 79 x 8

9. FINANCING RELATIONS (ADRIANO/CARL) – Finance.##h

Definition of Financing Relations:

Financial relations are defined as the provision of funds (legal and illegal) to support, sustain, and conduct operations for the terror network.

Types of Funding:

1. Money transfer from unknown source—in the form of cash or gold.
2. Crime—defined as illegal activities to raise funds, e.g. robberies.
3. Donations—*Infaq* and other collections of money.
4. Sales—the creation of a product as a mechanism to generate funds, e.g. the production of Videos or CDs.
5. Business—profit-based organization that uses some of its profits to support terror-related activities.

Two-mode matrix, 79 x 5

10. OPERATIONAL RELATIONS (CIOLPONEA/SCHUHART/EVERTON) – Ops.##h

Definition of Operational Relations:

Operational relations are defined as terrorists who are directly involved with the bombings, either at the scene e.g. a suicide bomber, commander or as a direct support to those at the scene e.g. driver or lookout. Matrix does not include communications, logistics, or organizational ties that were related to the operations. This file has been updated to reflect Koschade article on Bali I bombing.

List of Operations:

1. Australian Embassy Bombings
2. Bali Bombing I
3. Bali Bombing II
4. Marriott Bombings

Two-mode matrix 79 x 4

11. FRIENDSHIP RELATIONS (MAJED/TERRY) – Friendship.##h

Definition of Friendship Relations:

Friendship relations are defined as close attachments through affection or esteem between two people. Friendship ties are not defined as meetings and/ or school ties.

One-mode matrix, 79x79

12. RELIGIOUS TIES (MAJED/TERRY) – Religious.##h

Definition of Religious Relations:

Religious relations are defined as association with a mosque. We will not include Islamic schools, even though we assume that the schools have mosques. Not using the schools prevents duplication of effort with the team constructing the school ties. We listed the Mosques by the town in which it is located. If there was more than one in a city, we added a numerical identifier plus the name of nearest location.

List of Mosques:

1. Surabaya Mosque I (al – Ikhsan Mosque)
2. Surabaya Mosque II (Airlangga University)
3. Kediri Mosque
4. Banten Mosque
5. Cipayung Mosque
6. Solo Mosque (an – Nur Campus)

Two-mode matrix, 79 x 6

13. LOGISTICAL RELATIONS (BRIZEK, SWEENEY) – LogP.##h

Definition of Logistical Relations:

Logistical relations are defined to mean a Key Place within the archipelago where logistical activity occurred. Logistical activity is defined as providing “safe houses” for meeting/hiding, providing material support in terms of explosives, providing weaponry, or facilitating transportation of personnel or equipment.

List of Places Where Logistical Support Given:

1. Kuta
2. Boyolali
3. Kartosura
4. Pekalongan
5. Semarang
6. Solo
7. Surakarta
8. Ungaran
9. Blitar
10. Malang
11. Mojoagung
12. Mojokerto
13. Pasuruan
14. Surabaya
15. Ambon
16. Buru
17. Poso
18. Bukittinggi
19. Bengkulu
20. Dumai

21. Pekanbaru
22. Medan
23. Anyer
24. Jakarta
25. Bandung
26. Cianjur
27. Indramayu
28. Palabuhanratu
29. Tawau
30. Sekudai
31. Cotabato
32. Datu Piang
33. Zamboanga
34. Yogyakarta
35. Wonosobo

Two-mode matrix, 79 x 35

14. LOGISTICAL FUNCTIONS (BRIZEK, SWEENEY) – LogF.##h

Definition of Logistic Functions:

Logistical functions are defined as the support for terrorist operations by providing materials, weapons, transportation and safehouses.

List of Logistic Functions:

1. Safehouses
2. Weapons
3. Transportation
4. Material

Two-mode matrix, 79 x 4

15. ATTRIBUTES – Noordin Attribues.##h

(BELCHER/IANCU)

Education Level:

Defined as highest degree attained, level taught at, studied, participated in, or attended.

Coding Scheme:

0. Unknown
1. Elementary Education
2. Pesantren (Luqmanul Hakiem, Ngruki, al-Husein, Indramayu, Jemaah Islamiyah)
3. State High School
4. Some University (University an-Nur, Univeristi Teknologi Malaysia, Adelaide University, Bogor Agricultural Univ.)
5. BA/BS Designation
6. Some Graduate
7. Masters
8. PhD (Reading University)

Contact with People outside Indonesia:

Defined as contact with people in different countries outside Indonesia

Coding Scheme:

0. Unknown
1. Afghanistan
2. Australia
3. Malaysia
4. Pakistan
5. Philippines
6. Singapore
7. Thailand
8. United Kingdom
9. Afghanistan & Malaysia
10. Afghanistan & Pakistan
11. Afghanistan & Philippines
12. Afghanistan, Malaysia, & Philippines
13. Australia & Malaysia
14. Philippines & Malaysia

Military Training:

Defined the country where a terrorist received military training and attained veteran status in fighting in known insurgent/conventional wars:

Coding Scheme:

0. Unknown
1. Afghanistan
2. Australia
3. Indonesia
4. Malaysia
5. Philippines
6. Singapore
7. Afghanistan & Indonesia
8. Afghanistan & Philippines
9. Indonesia & Malaysia
10. Indonesia & Philippines

Nationality of terrorists:

Defined as country of birth, citizenship, or residence:

Coding Scheme:

1. Afghanistan
2. Australia
3. Indonesia
4. Malaysia
5. Philippines
6. Singapore

(JB/MATT)

Current Status per ICG Article:

Defined as the physical condition of the terrorist

Coding Scale:

1. dead
2. alive
3. jail

Role:

Defined as the role a terrorist assumes in the terror network

Coding Scheme:

0. no info / unclear
1. strategist: high level planner of a terror network
2. bomb maker: individual who constructs bombs
3. bomber/fighter: individual who participates in bombing attacks or who is described as a fighter
4. trainer/instructor: individual who trains or instructs new members of a terror network
5. suicide bomber: individual who plans to or already has performed a suicide attack
6. recon and surveillance – engaged in the surveillance and recon of targets
7. recruiter – engaged in identifying and recruiting new members (to include bombers)
8. courier /go-between – used in communications between members
9. propagandist – developed information campaigns
10. facilitator – assisted in the operation of the network (especially with materials and finances)
11. religious leader – provided religious training and support
12. commander/ tactical leader – in charge of operations at the local/tactical level

Logistics Function

Defined as the provision of safe houses, weapons, transportation, material to the operational network.

Coding scheme for the attribute:

1. Providing a safe house
2. Providing weapons
3. Providing transportation
4. Providing material
5. Providing weapons, transportation, material
6. Providing weapons, material
7. Providing transportation, material
8. Providing safehouse and transportation
9. Providing safehouse, transportation, material
10. Providing safehouse, weapons, material

(EVERTON/ABELL/DEFEYTER)

Current Status Updated:

Defined as the physical condition of the terrorist

Coding Scale:

0. Dead
1. Alive
2. Jail

(ABELL/DEFEYTER/EVERTON)

Effectiveness

Defined as the effectiveness of the terrorist with regards to Noordin's network

Coding Scale:

0. Dead
1. Free but on the run
2. Free but compromised
3. Free and active
4. Jail
5. Flipped (now a good guy)

APPENDIX 2

GLOSSARY OF TERMS

Actor: Actors can be people, subgroups, organizations, collectivities, communities, nation-states, etc. and are represented by a vertex in a social network.

Affiliation network: A type of *two-mode network* consisting of one set of actors and one set of events.

Arc: A *directed* line that connects one actor in a *digraph* (directed graph) to another actor.

Complete network: A *complete network* is a network with a density of one (i.e., maximum density).

Degree: The *degree* of a vertex equals the number of lines *incident* with it.

Density: *Density* is the number of lines in a simple network, expressed as a proportion of the maximum possible number of lines.

Dyadic network: A type of *two-mode network* consisting of two sets of actors.

Digraph (Directed graph): A graph where one or more lines (*arc*) are directed from one *vertex* to another.

Directed line: A *directed line* is commonly known as an *arc,* which is simply a line that points from one vertex to another.

Edges: An undirected line that connects one actor to another.

Fruchterman Reingold: The Fruchterman Reingold algorithm attempts to simulate a system of mass particles where the vertices simulate mass points repelling each other while the edges simulate springs with attracting forces. It then tries to minimize the "energy" of this physical system. It differs from the Kamada-Kawai algorithm in that it is able to distribute points in both two-

dimensional and three-dimensional space. See also *Kamada-Kawai* and *Spring embedded algorithms*.

Geodesic: Geodesic distance is the length of the shortest path between two nodes (actors).

Graph: A graph is a model for a social network with ties between pairs of *actors* (vertices). A tie can be either present or absent between each pair of *actors*. See *digraph* and *simple (directed and undirected) graph*.

Incident: A *line* is defined by its endpoints (vertices), which are said to be *incident* with the line.

Kamada-Kawai: The Kamada-Kawai spring embedded algorithm assumes an attraction between adjacent points (vertices), repulsion between non-adjacent points and allocates points in two-dimensional space. See also *Fruchterman Reingold* and *Spring embedded algorithms*.

Line: A line is a relation between two vertices (e.g., actors or events). They can be either *directed* or *undirected*.

Loop: A loop is a line that connects a vertex with itself.

Multidimensional scaling: Algorithms (see metric and non-metric multidimensional scaling) that allocate a social network's nodes in k-dimensional space (e.g., 2D, 3D).

Network: A network consists of a graph with additional information concerning the graph's vertices and/or lines.

Node: Vertex

Nodal degree: The degree of a *node* is the number of lines that are incident with it.

One-mode network: A network that consists of a single set of actors. See also *two-mode network*.

Partition: A network *partition* is a discrete classification or clustering of vertices that assigns each vertex to exactly one class or cluster.

Path: A path is a *walk* (i.e., a sequence of actors and ties) in which no actor in between the first and last actor of the walk occurs more than once.

Simple undirected graph: A *graph* that has no loops (i.e., a line between a *node* and itself) and includes no more than one *edge* between a pair of *nodes*.

Simple directed graph: A *graph* that does not contain multiple arcs (loops are allowed, however).

Spring embedded algorithms: Graph-drawing algorithms that treat points (vertices) as pushing and pulling on one another that seeks to find an optimum solution where there is a minimum amount of stress on the springs connecting the whole set of points. See also *Fruchterman Reingold* and *Kamada-Kawai*.

Two-mode network: A network that consists of two sets of actors (i.e., *dyadic network*), or one set of actors and one set of events (i.e., *affiliation network*). See also *one-mode network.*

Undirected line: An *undirected line* is a line that connects two vertices but does not point from one vertex to another.

Vector: In Pajek, a *vector* is a numerical (continuous) value assigned to each vertex in a network

Walk: A walk is a sequence of actors and ties that begins and ends with actors.

APPENDIX 3

MULTIDIMENSIONAL SCALING USING UCINET

This appendix illustrates how to estimate multidimensional (MDS) scaling coordinates in UCINET, which can then be read by NetDraw. The advantage of calculating MDS coordinates in UCINET is that UCINET calculates a stress statistic (while NetDraw currently does not). Stress statistics are valuable because they indicate how well the network map fits the data. As noted earlier, a stress statistic greater than .20 is generally considered intolerable. Thus, we would not want to rely too heavily on a network map with a high stress statistics. Another advantage of calculating coordinates in UCINET rather than in NetDraw is that with UCINET users can estimate metric and non-metric MDS coordinates, while NetDraw currently only offers metric MDS. For this appendix we will use relatively small social network data because it is easier to illustrate these techniques with smaller network than with larger ones. That said, there is nothing we do in this appendix that you cannot do with larger datasets.

A3.1 Multidimensional Scaling of Symmetric One-Mode Networks

We will begin with symmetric one-mode networks because it is easier to estimate MDS coordinates for symmetric one-mode networks than asymmetric ones. For this, we use the marital ties of Padgett's Florentine Families, which we have discussed elsewhere in this guide (see Figure 1.1 above). Our first task is to use this network to calculate a set of related coordinates. We consider both metric and non-metric MDS. We will then read these (and the related network) into NetDraw.

Metric Multidimensional Scaling in UCINET

As we noted earlier, network analysts have long used sociograms to visualize social networks, and in recent years analysts have begun using a series of mathematical techniques to locate the points of a network in such a way that the distances between them are meaningful. MDS is one such technique. It uses the concepts of space and distance to represent a network's internal structure (Wasserman and Faust 1994). The typical input is a symmetric matrix consisting of measures of similarity or dissimilarity between pairs of actors. Output generally consists of a set of estimated distances between pairs of actors that we

can represent in one-, two-, three- or higher-dimensional space (Kruskal and Wish 1978; Wasserman and Faust 1994).

Metric MDS takes a given matrix of proximities that measure the similarities or dissimilarities among a set of actors and calculates a set of points in k-dimensional space, such that the distances between them correspond as closely as possible to the input proximities (Borgatti, Everett and Freeman 1999).[54] Metric distance differs from distance in graph theory. In graph theory, the distance between two points is measured in terms of the number of lines in the path that connects the two points. In MDS the distance between two points is the most direct route between them. "It is a distance that follows a rout 'as the crow flies', and that may be across 'open space' and need not – indeed, it normally will not – follow a graph theoretical path" (Scott 2000:148-149).

Tools >Scaling/Decomposition >Metric MDS

Under UCINET's *Tools* menu, select the *Scaling/Decomposition>Metric MDS* command; this should bring up a dialog box similar to Figure A3.1. There are a number of options available. In general, you will want to accept UCINET's defaults unless you have a good reason not to. Here, I changed only one: the name of the output dataset in order to make it easier to identify.

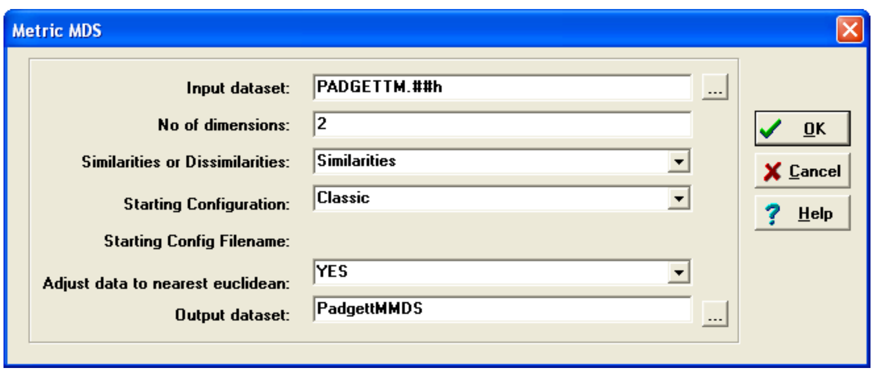
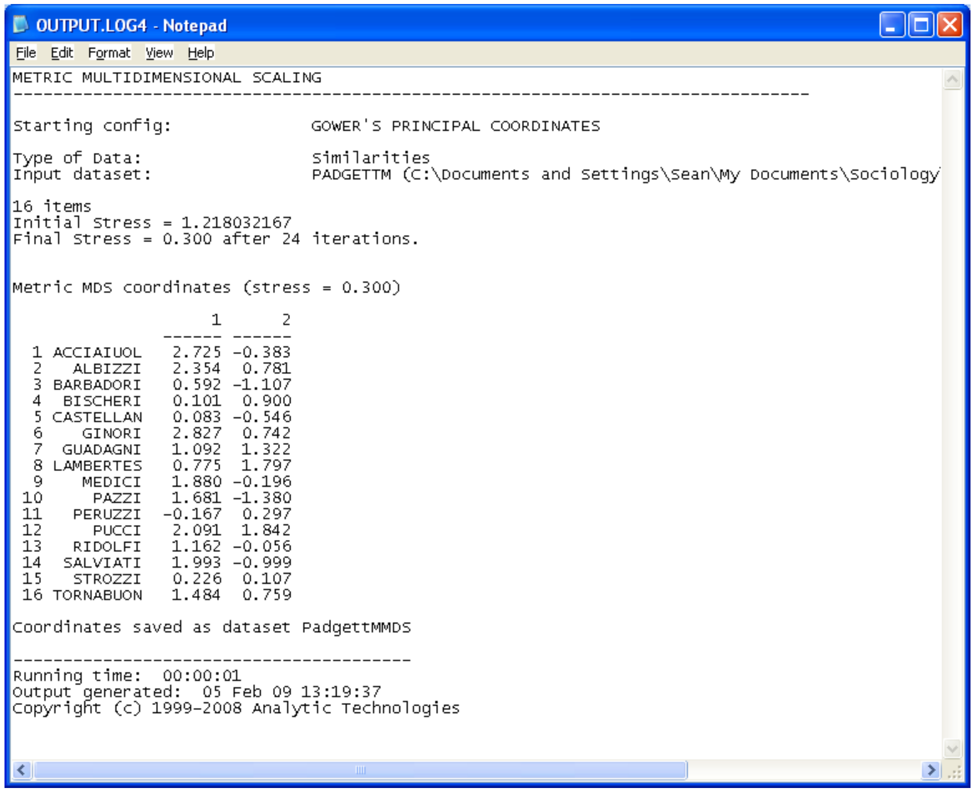


Figure A3.1: Metric MDS Dialog Box

Running this procedure produces both a scatterplot (which we do not need) and an output file that lists the MDS coordinates and a stress score (Figure A3.2, next page). As you can see, the stress is .300, which tells us that the coordinates do not fit the data relatively well. The coordinates themselves are stored in the file PadgetMMDS, which we will use later use to export to Mage. Before seeing how to do that, however, let us first see what happens when compute coordinates using Nonmetric MDS.

[54] The Padgett data proximities represent similarities between the families. That is, a "1" in a matrix cell means that the two families represented by that cell share a marital tie.



```
OUTPUT.LOG4 - Notepad
File  Edit  Format  View  Help
METRIC MULTIDIMENSIONAL SCALING
--------------------------------------------------------------------------------

Starting config:              GOWER'S PRINCIPAL COORDINATES

Type of Data:                 Similarities
Input dataset:                PADGETTM (C:\Documents and Settings\Sean\My Documents\Sociology

16 items
Initial Stress = 1.218032167
Final Stress = 0.300 after 24 iterations.


Metric MDS coordinates (stress = 0.300)

                     1      2
                ------ ------
  1 ACCIAIUOL    2.725 -0.383
  2   ALBIZZI    2.354  0.781
  3 BARBADORI    0.592 -1.107
  4  BISCHERI    0.101  0.900
  5 CASTELLAN    0.083 -0.546
  6    GINORI    2.827  0.742
  7  GUADAGNI    1.092  1.322
  8 LAMBERTES    0.775  1.797
  9    MEDICI    1.880 -0.196
 10     PAZZI    1.681 -1.380
 11   PERUZZI   -0.167  0.297
 12     PUCCI    2.091  1.842
 13   RIDOLFI    1.162 -0.056
 14  SALVIATI    1.993 -0.999
 15   STROZZI    0.226  0.107
 16 TORNABUON    1.484  0.759

Coordinates saved as dataset PadgettMMDS

----------------------------------------
Running time:  00:00:01
Output generated:  05 Feb 09 13:19:37
Copyright (c) 1999-2008 Analytic Technologies
```

Figure A3.2: UCINET's Metric MDS Output

Nonmetric Multidimensional Scaling in UCINET

As noted earlier there are some limitations to using metric MDS for visualizing social networks. Many relational data sets, such as the Padgett data, are binary in form in that they simply indicate either the presence or absence of a tie. As a consequence, we cannot directly use such data to measure proximities. We first need to convert it into other measures, such as correlation coefficients, before calculating its metric properties. However, because the data are not metric (i.e., they only consist of 1's and 0's) we can possibly draw incorrect conclusions about the data. Even when the data are valued, metric assumptions may be inappropriate. For example, a family with four marital ties may not be twice as central a family with only two.

While nonmetric MDS procedures (like metric MDS procedures) use symmetrical adjacency matrices to calculate similarities or dissimilarities between actors, unlike metric MDS, they do not convert these values directly into Euclidean distances. Instead, they consider only rank order. They treat the data, in other words, as ordinal. They "seek a solution in which the rank ordering of the distances is the same as the rank ordering of the original values" (Scott 2000:157).

Non-metric MDS is often preferred because it tends to provide a better "goodness-of-fit" (stress) statistic.

[UCINET] Tools >Scaling/Decomposition >Non-metric MDS

To estimate nonmetric MDS coordinates, select the *Non-metric MDS* option found under the *Tools>Scaling/Decomposition* submenu. This brings up a dialog box similar to the one we used for calculating metric MDS (Figure A3.3). As before, I accepted all of UCINET's defaults except that I changed the name of the output dataset in order to make it easier to identify.

Figure A3.3: UCINET Non-metric MDS Scaling Dialog Box

This procedure also produces both a scatterplot and an output file that lists the MDS coordinates and a stress score (Figure A3.4, next page). As you can see, the stress is .200, which is a better fit than we got with metric MDS. It is not ideal, but it is at least within an acceptable range. Thus, we will use the nonmetric coordinates (PadgettNMDS) for visualizing in NetDraw.

Using UCINET Coordinates in NetDraw

[NetDraw] File>Open >Ucinet dataset>Network

File>Open >Ucinet dataset >Coordinates

Using MDS coordinates in NetDraw is straightforward. To do so, first open the Padgett marriage data in NetDraw with its *File>Open>Ucinet dataset> Network* command. Next open the related coordinate file with the command *File>Open>Ucinet dataset>Coordinates.* Net Draw automatically assigns these coordinates to the respective nodes. You should get a network map similar to the one displayed in Figure A3.5. If you choose a mapping algorithm resident in NetDraw, then in order to recall the coordinates calculated in UCINET, you will need to issue the *File>Open>Ucinet dataset>Coordinates* command again. An interesting experiment would be to compare this network map to one that uses one of NetDraw's mapping algorithms.

```
OUTPUT.LOG5 - Notepad
File  Edit  Format  View  Help
NON-METRIC MDS
--------------------------------------------------------------------------------

Starting config:              TORSCA
Type of Data:                 Similarities

Input dataset:                PADGETTM (C:\Documents and Settings\Sean\My Documents\Sociology

Non-metric MDS coordinates (stress = 0.020)

                    1       2
              ------- -------
    ACCIAIUOL   0.811   0.865
      ALBIZZI   0.212   0.381
    BARBADORI  -0.377   0.020
     BISCHERI   0.397  -1.010
    CASTELLAN  -0.283  -0.835
       GINORI  -0.451   0.711
     GUADAGNI   0.086  -0.343
    LAMBERTES  -0.769  -0.448
       MEDICI   0.299   0.347
        PAZZI  -0.349   1.631
      PERUZZI  -0.167  -1.323
        PUCCI  -1.528   0.222
      RIDOLFI   0.762  -0.253
     SALVIATI   0.138   1.085
      STROZZI   0.446  -0.934
    TORNABUON   0.774  -0.116

Stress = 0.020 in 50 iterations.

----------------------------------------
Running time:  00:00:01
Output generated:  05 Feb 09 13:22:23
Copyright (c) 1999-2008 Analytic Technologies
```

Figure A3.4: UCINET's Non-Metric MDS Output

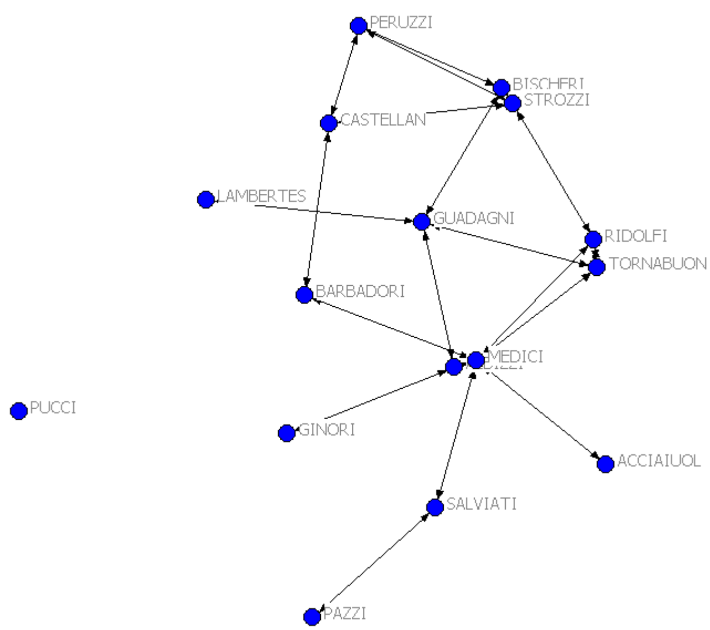


Figure A3.5: Non-metric MDS Map of Padgett Marriage Network

A3.2 Visual Representation of Asymmetric One-Mode Networks

Visualizing asymmetric (directional) one-mode networks using UCINET and Mage is somewhat different because the MDS routines require symmetric matrices (although you can submit asymmetric matrices to UCINET's *Tools>Scaling/ Decomposition>Metric MDS* and *Non-Metric MDS* commands and UCINET will output coordinates without a warning that you are not supposed to do so). Thus, we need to first calculate an equivalence matrix, based either on the distances (e.g., Euclidean) or the correlations between the nodes of the directed matrix. We then submit this equivalence matrix, which is symmetric, to MDS algorithms. For our purposes here we will use the advice network of Krackhardt's High-Tech Managers (1987), which we have used previously (see, e.g., Figures 3.2 and 3.3).

[UCINET] Network>Roles & Positions >Structural>Profile

To calculate an equivalence matrix, under the *Network* menu, choose the *Roles & Positions>Structural>Profile* command. This brings up UCINET's profile similarity dialog box (see Figure A3.6 below). As is generally the case, accept UCINET's defaults although change the "Measure of profile similarity/ distance" option to Correlation since in this case a tie between two individuals indicates similarity. You will also want to change the names of the output files in order to make them easier to identify. This produces both a dendogram (not shown) and a structural equivalence matrix (Figure A3.7, next page).

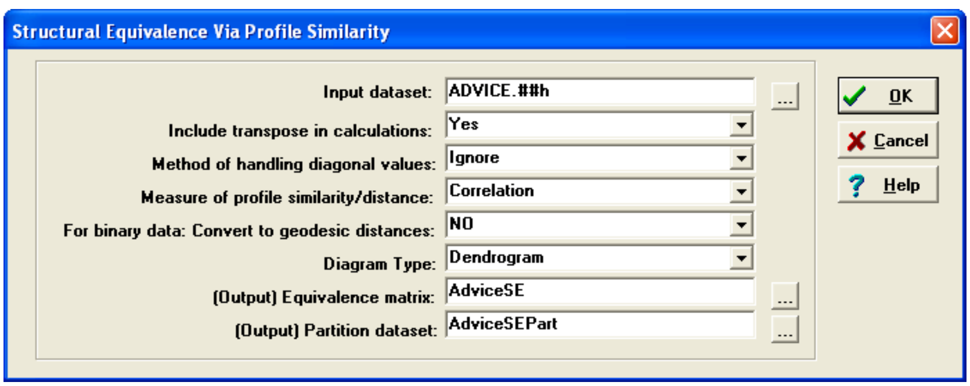
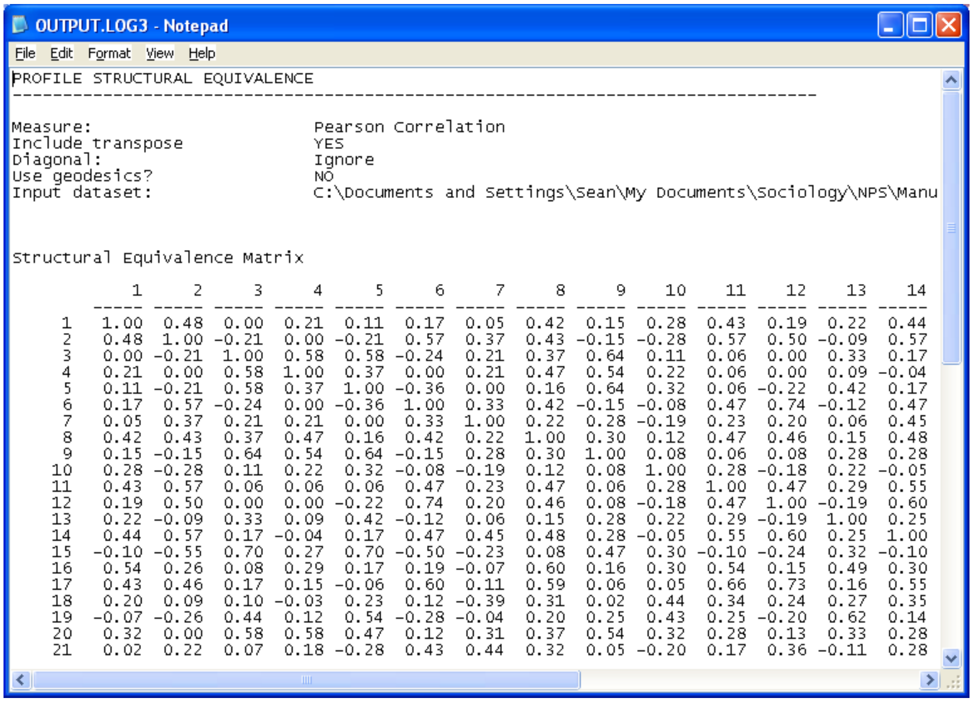


Figure A3.6: UCINET Structural Equivalence Profile Dialog Box

The next step in the process is to submit the structural equivalence matrix to the MDS techniques discussed earlier (not shown).[55] The stress statistics for

[55] If we had chosen to calculate the matrix using the Euclidean distance option, then in the resulting matrix the larger the number would indicate the greater the distance of one actor from another. For example, in Figure A3.7 the correlation coefficients along the diagonal are all 1.00 (because each actor is perfectly correlated with itself); if we had chosen the Euclidean distance option, the coefficients along the diagonal would be 0.00. Thus, if we hand chosen the Euclidean option, when we instructed UCINET to perform MDS on the structural equivalence matrix, we would need to choose the "Dissimilarities" option rather than the "Similarities" option (see Figure A3.8).

metric and nonmetric MDS were .196 and .129 respectively; thus we uses the nonmetric MDS coordinates for our network map in NetDraw (See Figure A3.9).



OUTPUT.LOG3 - Notepad

File Edit Format View Help

```
PROFILE STRUCTURAL EQUIVALENCE
--------------------------------------------------------------------------------

Measure:                         Pearson Correlation
Include transpose                YES
Diagonal:                        Ignore
Use geodesics?                   NO
Input dataset:                   C:\Documents and Settings\Sean\My Documents\Sociology\NPS\Manu
```

Structural Equivalence Matrix

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	1.00	0.48	0.00	0.21	0.11	0.17	0.05	0.42	0.15	0.28	0.43	0.19	0.22	0.44
2	0.48	1.00	-0.21	0.00	-0.21	0.57	0.37	0.43	-0.15	-0.28	0.57	0.50	-0.09	0.57
3	0.00	-0.21	1.00	0.58	0.58	-0.24	0.21	0.37	0.64	0.11	0.06	0.00	0.33	0.17
4	0.21	0.00	0.58	1.00	0.37	0.00	0.21	0.47	0.54	0.22	0.06	0.00	0.09	-0.04
5	0.11	-0.21	0.58	0.37	1.00	-0.36	0.00	0.16	0.64	0.32	0.06	-0.22	0.42	0.17
6	0.17	0.57	-0.24	0.00	-0.36	1.00	0.33	0.42	-0.15	-0.08	0.47	0.74	-0.12	0.47
7	0.05	0.37	0.21	0.21	0.00	0.33	1.00	0.22	0.28	-0.19	0.23	0.20	0.06	0.45
8	0.42	0.43	0.37	0.47	0.16	0.42	0.22	1.00	0.30	0.12	0.47	0.46	0.15	0.48
9	0.15	-0.15	0.64	0.54	0.64	-0.15	0.28	0.30	1.00	0.08	0.06	0.08	0.28	0.28
10	0.28	-0.28	0.11	0.22	0.32	-0.08	-0.19	0.12	0.08	1.00	0.28	-0.18	0.22	-0.05
11	0.43	0.57	0.06	0.06	0.06	0.47	0.23	0.47	0.06	0.28	1.00	0.47	0.29	0.55
12	0.19	0.50	0.00	0.00	-0.22	0.74	0.20	0.46	0.08	-0.18	0.47	1.00	-0.19	0.60
13	0.22	-0.09	0.33	0.09	0.42	-0.12	0.06	0.15	0.28	0.22	0.29	-0.19	1.00	0.25
14	0.44	0.57	0.17	-0.04	0.17	0.47	0.45	0.48	0.28	-0.05	0.55	0.60	0.25	1.00
15	-0.10	-0.55	0.70	0.27	0.70	-0.50	-0.23	0.08	0.47	0.30	-0.10	-0.24	0.32	-0.10
16	0.54	0.26	0.08	0.29	0.17	0.19	-0.07	0.60	0.16	0.30	0.54	0.15	0.49	0.30
17	0.43	0.46	0.17	0.15	-0.06	0.60	0.11	0.59	0.06	0.05	0.66	0.73	0.16	0.55
18	0.20	0.09	0.10	-0.03	0.23	0.12	-0.39	0.31	0.02	0.44	0.34	0.24	0.27	0.35
19	-0.07	-0.26	0.44	0.12	0.54	-0.28	-0.04	0.20	0.25	0.43	0.25	-0.20	0.62	0.14
20	0.32	0.00	0.58	0.58	0.47	0.12	0.31	0.37	0.54	0.32	0.28	0.13	0.33	0.28
21	0.02	0.22	0.07	0.18	-0.28	0.43	0.44	0.32	0.05	-0.20	0.17	0.36	-0.11	0.28

Figure A3.8: UCINET Structural Equivalence (Correlation) Matrix

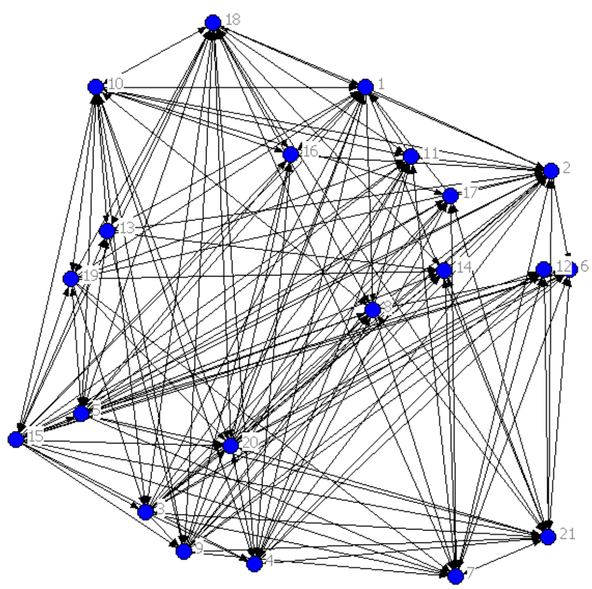


Figure A3.9: Non-metric MDS Map of Krackhardt Advice Network

A3.3 Visual Representation of Two-Mode Networks

Two-mode data present additional visualization complexities. To illustrate these, we will use Davis's Southern Club Women (Breiger 1974; Davis, Gardner and Gardner 1941), which we first considered in Chapter 4 (Section 4.3 – see Figure 4.3). Depending on how we manipulate the data, we can use UCINET to visualize two-mode networks in a variety of ways. We can, of course, convert the data to one-mode (actors or events) data and visualize them using the techniques discussed in Section A3.1. Alternatively, we can visualize the original two-mode network. That is the focus of this section. There are a number of approaches (Borgatti and Everett 1997; Everton 2004). A common approach is to use correspondence analysis; however, Borgatti and Everett (1997:247) argue that in correspondence analysis representations of two-mode data is that the distances between nodes are not Euclidean (i.e., the distances do not necessarily reflect social distance) (see, however, Roberts 2000). As such, they recommend that we first convert the two-mode network to a bipartite graph (explained below), from which we compute the *geodesic* distances between all pairs of nodes, which we then submit to MDS techniques (Borgatti and Everett 1997:249-251). This approach is illustrated below.

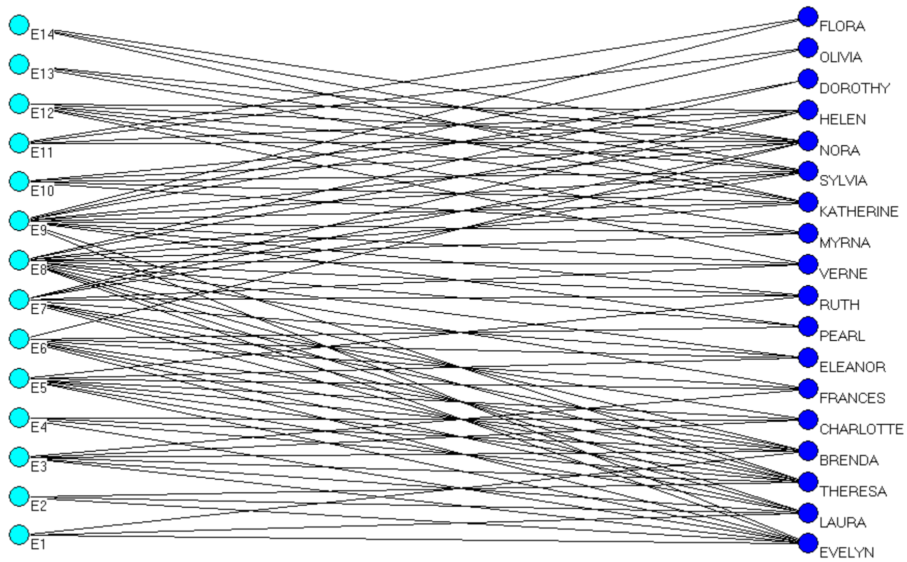


Figure A3.10: Visual Representation of Bi-Partite Graph

Multidimensional Scaling of Two-Mode Data

What is a bipartite graph? "A graph is bipartite if the vertices may be partitioned in exactly two mutually exclusive sets such that there are no ties

wholly within either set – i.e., the endpoints of every tie come from different sets" (Borgatti and Everett 1997:247-248) (See Figure A3.10, previous page).

[UCINET] Transform>Bipartite

To create a bi-partite graph from a two-mode network we use UCINET's *Transform>Bipartite* command. This brings up a dialog box (Figure A3.11) that requires us to indicate which two-mode network we want to transform. It is important to note that you will want to tell UCINET to make the resulting graph *symmetric* – this is not UCINET's default option. If you do not change the option to symmetric, you will not be able to calculate the geodesic distance between nodes in the following step. Clicking OK will generate a bi-partite graph/matrix (not shown) that upon close inspection should be a symmetric, one-mode graph with 32 rows and columns (18 women + 14 events).

Figure A3.11: Bi-Partite Dialog Box

[UCINET] Network>Cohesion >Distance

Recall that geodesic distance refers to the length of the shortest path between two nodes. To calculate this in UCINET we use the *Network>Cohesion> Distance* command. In the resulting dialog box (Figure A3.12) indicate that the input dataset is the bipartite network calculated earlier, accept UCINET's defaults and click OK.

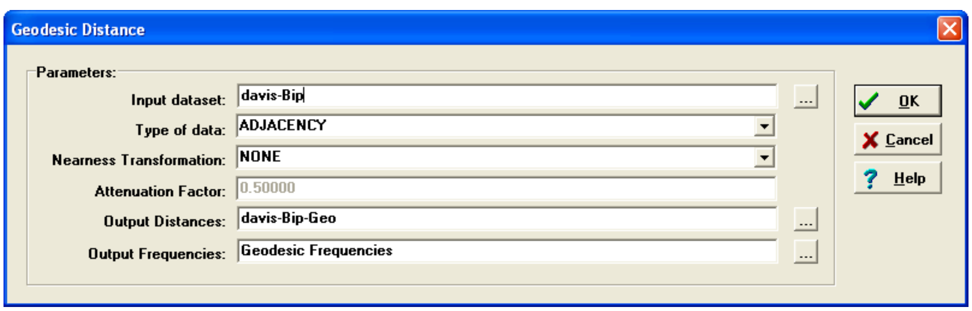


Figure A3.12: Geodesic Distance Dialog Box

This produces a distance matrix (not shown). If you examine the matrix closely you will note that the geodesic distances between any two women or between any two events is never less than two (Borgatti and Everett 1997:249; Faust 1997). This is because the women are only connected to one another through events and the events are only connected to one another through women, so there it always takes at least two steps to get from one woman to another or from one event to another.

The next step is submitting this distance matrix to the MDS routines discussed earlier. Because metric MDS yielded a stress test of .348, while nonmetric MDS yielded a stress statistic of .213, we use the nonmetric MDS coordinates after opening the original *Southern Women* dataset (not the geodesic distance matrix) in NetDraw, which yields a network map similar to Figure A3.14. Of course, since the stress statistic is above .200, we should take this network map with a grain of salt, recognizing the it does not fit the data too well.

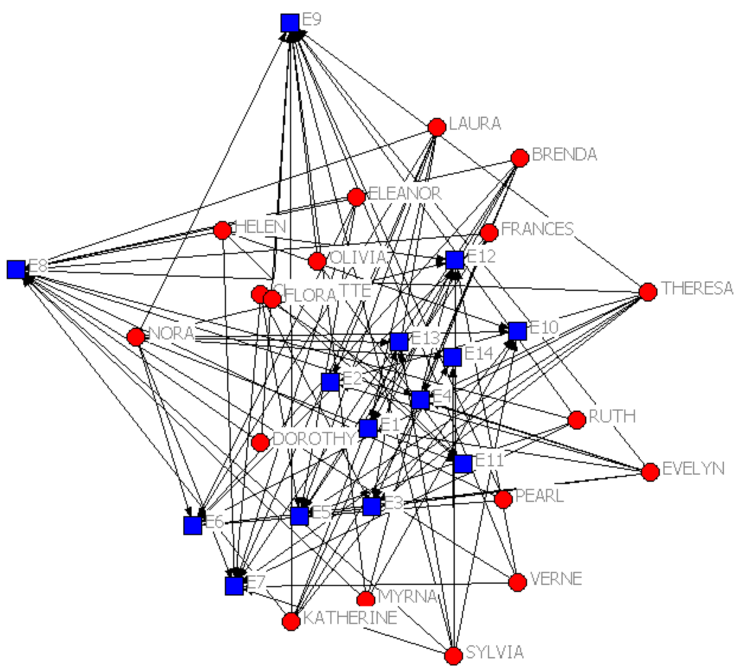


Figure A3.13 Geodesic MDS Network Map of Southern Women Data

References

Arreguin-Toft, Ivan. 2005. *How the Weak Win Wars: A Theory of Asymmetric Conflict*. Cambridge, UK: Cambridge University Press.

Asch, Solomon E. 1955. "Opinions and Social Pressure." *Scientific American* 193:31-35.

Azarian, G. Reza. 2005. *The General Sociology of Harrison C. White: Chaos and Order in Networks*. New York: Palgrave Macmillan.

Barabasi, Albert-Laszlo. 2002. *Linked: The New Science of Networks*. Cambridge, MA: Perseus Publishing.

Barabasi, Albert-Laszlo, and Reka Albert. 1999. "Emergence of Scaling in Random Networks." *Science* 286:509-512.

Barabasi, Albert-Laszlo, Reka Albert, and Hawoong Jeong. 1999. "Mean-Field Theory for Scale-Free Random Networks." *Physica A* 272 (173-187).

Barabasi, Albert-Laszlo, and Eric Bonabeau. 2003. "Scale-Free Networks." *Scientific American* 288 (5):60-69.

Barnes, John A. 1954. "Class and Committee in a Norwegian Island Parish." *Human Relations* 7:39-58.

Batagelj, Vladimir, and Andrej Mrvar. 2007. *Pajek 1.20*. Lubjlijana, Slovenia: University of Ljubljana.

Bauerlein, Mark. (2004, November 12). Liberal Groupthink is Anti-Intellectual. *The Chronicle of Higher Education,* pp. B6-B10.

Bavelas, Alex. 1950. "Communication Patterns in Task-Oriented Groups." *Journal of the Acoustical Society of America* 22:725-730.

Bonacich, Phillip. 1987. "Power and Centrality: A Family of Measures." *American Journal of Sociology* 92 (5):1170-1182.

Borer, Douglas A. 2008. "The Strategy-Legitimacy Nexus: An Essay on Iraq." Paper presented at Modern Protectorates, Culture and Dark Networks, Konstanz, Germany.

Borgatti, Stephen P. 2002-2005. *NetDraw 2.0*. Harvard, MA: Analytical Technologies.

Borgatti, Stephen P., and Martin G. Everett. 1997. "Network Analysis of 2-mode Data." *Social Networks* 19:243-269.

Borgatti, Stephen P., Martin G. Everett, and Linton C. Freeman. 1999. *Ucinet 5 for Windows: Software for Social Network Analysis*. Natick: Analytical Technologies.

________. 2002. *Ucinet for Windows: Software for Social Network Analysis*. Harvard, MA: Analytical Technologies.

Bott, Elizabeth. 1957. *Family and Social Network*. London, England: Tavistock.

Brafman, Ori, and Rod A. Beckstrom. 2006. *The Starfish and the Spider: The Unstoppable Power of Leaderless Organizations*. New York: Portfolio.

Breiger, Ronald L. 1974. "The Duality of Persons and Groups." *Social Forces* 53:181-190.

________. 1975. *Dual and Mutliple Networks of Social Structures: A Study of Affiliation and Interaction*. Doctoral Dissertation, Sociology. Harvard University.

Breiger, Ronald L., and Philippa E. Pattison. 1986. "Cumulated Social Roles: The Duality of Persons and their Algebras." *Social Networks* 8:215-256.
Buchanan, Mark. 2001. *Ubiquity*. New York: Crown Publishers.
________. 2002. *Nexus: Small Worlds and the Groundbreaking Science of Networks*. New York: W. W. Norton & Company.
Burt, Ronald S. 1984. "Network Items and the General Social Survey." *Social Networks* 6:293-340.
________. 1985. "General Social Survey Network Items." *Connections* 8:119-122.
________. 1992a. "The Social Structure of Competition." Pp. 57-91 in *Networks and Organizations: Structure, Form and Action*, edited by Nitin Nohria and Robert G. Eccles. Boston: Harvard University Press.
________. 1992b. *Structural Holes: The Social Structure of Competition*. Cambridge, MA: Harvard University Press.
Carley, Kathleen M., Ju-Sung Lee, and David Krackhardt. 2002. "Destabilizing Networks." *Connections* 24 (3):79-92.
Carley, Kathleen M., Jeffrey Reminga, and Natasha Kamneva. 2003. "Destabilizing Terrorist Networks." Paper presented at 8th International Command and Control Research and Technology Symposium, National Defense War College, Washington DC.
Castilla, Emilio, Hokyu Hwang, Ellen Granovetter, and Mark Granovetter. 2000. "Social Networks in Silicon Valley." Pp. 218-247 in *The Silicon Valley Edge: A Habitat for Innovation and Entrepreneurship*, edited by Chong-Moon Lee, Henry S. Rowen, William F. Miller, and Marguerite Gong Hancock. Stanford, CA: Stanford University Press.
Chaves, Mark. 1997. *Ordaining Women: Culture and Conflict in Religious Organizations*. Cambridge, MA: Harvard University Press.
Cook, Karen S., and Richard M. Emerson. 1978. "Power, Equity and Commitment in Exchange Networks." *American Sociological Review* 43:712-739.
Cook, Karen S., Richard M. Emerson, Mary R. Gilmore, and Toshio Yamagishi. 1983. "The Distribution of Power in Exchange Networks." *American Journal of Sociology* 89:275-305.
Cook, Karen S., Mary R. Gillmore, and Toshio Yamagishi. 1986. "Point and Line Vulnerability As Bases For Predicting the Distribution of Power in Exchange Networks: Reply to Willer." *American Journal of Sociology* 92 (2):445-448.
Cook, Karen S., and J. M. Whitmeyer. 1992. "Two Approaches to Social Structure: Exchange Theory and Network Analysis." *Annual Review of Sociology* 18:109-127.
CTC. 2006. "Harmony and Disharmony: Exploiting al-Qa'ida's Organizational Vulnerabilities." Combating Terrorism Center at West Point, West Point, NY.
Davis, Allison, Burleigh B. Gardner, and Mary R. Gardner. 1941. *Deep South: A Social-Anthropological Study of Caste and Class*. Chicago: University of Chicago Press.

de Nooy, Wouter, Andrej Mrvar, and Vladimir Batagelj. 2005. *Exploratory Social Network Analysis with Pajek, Structural Analysis in the Social Sciences*. Cambridge, UK: Cambridge University Press.
Defeyter, Bruce, and Lauren Abell. 2008. "Divide-Distrust-Dissolve: A 3-Dimensional Approach to Deactivating a Terrorist Network." Pp. 24. Monterey, CA: Naval Postgraduate School.
Degenne, Alain, and Michael Forse'. 1999. *Introducing Social Networks*. Translated by Arthur Borges. Thousand Oaks, CA: Sage Publications.
Diani, Mario. 2002. "Network Analysis." Pp. 173-230 in *Methods of Social Movement Research*, edited by Bert Klandermans and Suzanne Staggenborg. Minneapolis, MN: Univeristy of Minnesota Press.
DiMaggio, Paul J., and Walter W. Powell. 1983. "The Iron Cage Revisited: Institutional Isomorphism and Collective Rationality in Organizational Fields." *American Sociological Review* 48 (2):147-160.
Durkheim, Emile. 1951. *Suicide: A Study in Sociology*. Translated by John A. Spaulding and George Simpson. New York: The Free Press.
Emerson, Richard M. 1962. "Power-Dependence Relations." *American Sociological Review* 27 (1):31-41.
________. 1972a. "Exchange Theory, Part I: A Psychological Basis for Social Exchange." Pp. 38-57 in *Sociological Theories in Progress*, edited by Joseph Berger, Morris Zelditch, and B. Anderson. Boston: Houghton-Mifflin.
________. 1972b. "Exchange Theory, Part II: Exchange Relations and Network Structures." Pp. 58-87 in *Sociological Theories in Progress*, edited by Joseph Berger, Morris Zelditch, and B. Anderson. Boston: Houghton-Mifflin.
________. 1976. "Social Exchange Theory." Pp. 335-362 in *Annual Review of Sociology*. Palo Alto, CA: Annual Reviews Inc.
Everton, Sean F. 2004. *A Guide for the Visually Perplexed: Visually Representing Social Networks*. .41 ed. Stanford, CA: Stanford University.
Faust, Katherine. 1997. "Centrality in Affiliation Networks." *Social Networks* 19:157-191.
Fernandez, Roberto M., and Roger V. Gould. 1994. "A Dilemma of State Power: Brokerage and Influence in the National-Health Policy Domain." *American Journal of Sociology* 99 (6):1455-1491.
Finke, Roger, and Rodney Stark. 2005. *The Churching of America, 1776-2005: Winners and Losers in Our Religious Economy*. 2nd ed. New Brunswick, NJ: Rutgers University Press.
Frank, David John, Ann Hironaka, and Evan Schofer. 2000. "The Nation-State and the Natural Environment over the Twentieth Century." *American Sociological Review* 65 (1):96-116.
Freeman, Linton C. 1979. "Centrality in Social Networks I: Conceptual Clarification." *Social Networks* 1:215-239.
________. 1999. "Using Molecular Modeling Software in Social Network Analysis: A Practicum." University of California, Irvine, Irvine, CA. Retrieved June 15, 2000 (http://eclectic.ss.uci.edu/~lin/chem.html).

________. 2000. "Visualizing Social Networks." *Journal of Social Structure* 1 (1).
________. 2004. *The Development of Social Network Analysis: A Study in the Sociology of Science*. Vancouver, Canada: Empirical Press.
Freeman, Linton C., Cynthia M. Webster, and Deirdre M. Kirke. 1998. "Exploring Social Structure Using Dynamic Three-Dimensional Color Images." *Social Networks* 20:109-118.
Fruchterman, T., and E. Reingold. 1991. "Graph Drawing by Force-Directed Replacement." *Software--Practice and Experience* 21 (11):1129-1164.
Glock, Charles Y. 1964. "The Role of Deprivation in the Origin and Evolution of Religious Groups." Pp. 24-36 in *Religion and Social Conflict*, edited by Robert Lee and Martin E. Marty. New York: Oxford University Press.
Gould, Roger V. 1991. "Multiple Networks and Mobilization in the Paris Commune, 1871." *American Sociological Review* 56:716-729.
________. 1993. "Collective Action and Network Structure." *American Sociological Review* 58 (1):182-196.
Gould, Roger V., and Roberto M. Fernandez. 1989. "Structures of Mediation: A Formal Approach to Brokerage in Transaction Networks." *Sociological Methodology*:89-126.
Granovetter, Mark. 1973. "The Strength of Weak Ties." *American Journal of Sociology* 73 (6):1360-1380.
________. 1974. *Getting a Job*. Cambridge, MA: Harvard University Press.
________. 1979. "The Theory-Gap in Social Network Analysis." Pp. 501-518 in *Perspectives on Social Network Research*, edited by Paul W. Holland and Samuel Leinhardt. New York: Academic Press.
________. 1985. "Economic Action and Social Structure: The Problem of Embeddedness." *American Journal of Sociology* 91:481-510.
________. 2005. "The Impact of Social Structure on Economic Outcomes." *Journal of Economic Perspectives* 19 (1):33-50.
Hanneman, Robert A., and Mark Riddle. 2005. *Introduction to Social Network Methods*. University of California, Riverside. Available on the World Wide Web: http://faculty.ucr.edu/~hanneman/nettext/
Heider, Fritz. 1977. "Attitudes and Cognitive Organization." *Journal of Psychology* 21:107-112.
Huisman, Mark, and Marijtje A. van Duijn. 2005. "Software for Social Network Analysis." Pp. 270-316 in *Models and Methods in Social Network Analysis*, edited by Peter J. Carrington, John Scott, and Stanley Wasserman. Cambridge and New York: Cambridge University Press.
International Crisis Group. 2006. "Terrorism in Indonesia: Noordin's Networks." Asia Report #114. International Crisis Group, Brussels, Belgium.
Kamada, T., and S. Kawai. 1989. "An Algorithm for Drawing General Undirected Graphs." *Information Processing Letters* 31:7-15.
Klerks, Peter. 2001. "The Network Paradigm Applied to Criminal Organisations: Theoretical Nitpicking or a Relevant Doctrine for Investigators? Recent Developments in the Netherlands." *Connections* 24 (3):53-65.
Knoke, David, and Edward O. Laumann. 1982. "The Social Structure of National Policy Domains: An Exploration of Some Structural Hypothesis." Pp. 255-

270 in *Social Structure and Network Analysis*, edited by Peter V. Marsden and Nan Lin. Beverly Hills, CA: Sage.

Knoke, David, and Song Yang. 2007. *Social Network Analysis*. 2nd ed. Thousand Oaks, CA: Sage Publications, Inc.

Krackhardt, David. 1987. "Cognitive Social Structures." *Social Networks* 9:109-134.

________. 1992. "The Strength of Strong Ties: The Importance of Philos in Organizations." Pp. 216-239 in *Networks and Organizations: Structure, Form and Action*, edited by Nitin Nohria and Robert G. Eccles. Boston: Harvard University Press.

Krebs, Valdis. 2001. "Mapping Networks of Terrorist Cells." *Connections* 24 (3):43-52.

Kruskal, Joseph B., and Myron Wish. 1978. *Multidimensional Scaling*. Newbury Park, CA: Sage Publications.

Laumann, Edward O., Peter V. Marsden, and David Presnsky. 1983. "The Boundary-Specification Problem in Network Analysis." Pp. 18-34 in *Applied Network Analysis*, edited by Ronald S. Burt and Michael Minor. Beverly Hills, CA: Sage.

________. 1989. "The Boundary-Specification Problem in Network Analysis." Pp. 61-87 in *Research Methods in Social Network Analysis*, edited by Linton C. Freeman, Douglas R. White, and A. K. Romney. Beverly Hills, CA: Sage.

Lewin, Kurt. 1951. *Field Theory in the Social Sciences*. New York: Harper.

Lofland, John. 1977. "'Becoming A World-Saver' Revisited." *American Behavioral Scientist* 20:805-818.

Lofland, John, and Rodney Stark. 1965. "Becoming a World-Saver: A Theory of Conversion to a Deviant Perspective." *American Sociological Review* 30:862-875.

Loomis, Charles P., Julio O. Morales, Roy A. Clifford, and Olen E. Leonard. 1953. *Turrialba: Social Systems and the Introduction of Change*. Gelncoe, IL: The Free Press.

Mayo, Elton. 1933. *The Human Problems of an Industrial Civilization*. Cambridge, MA: Macmillan.

________. 1945. *The Social Problems of an Industrial Civilization*. London: Routledge and Kegan Paul.

McAdam, Doug. 1982. *Political Process and the Development of Black Insurgency, 1930-1970*. Chicago: University of Chicago Press.

________. 1986. "Recruitment to High Risk Activism: The Case of Freedom Summer." *American Journal of Sociology* 92:64-90.

________. 1988. "Micromobilization Contexts and Recruitment to Activism." *International Social Movement Research* 1:125-154.

________. 1999. *Political Process and the Development of Black Insurgency, 1930-1970*. 2nd ed. Chicago: University of Chicago Press.

McAdam, Doug, John D. McCarthy, and Mayer N. Zald. 1988. "Social Movements." Pp. 695-737 in *The Handbook of Sociology*, edited by Neil J. Smelser. Beverly Hills, CA: Sage.

________. 1996. "Introduction: Opportunities, Mobilizing Structures, and Framing Processes -- Toward a Synthetic, Comparative Perspective on Social Movements." in *Comparative Perspectives on Social Movements: Political Opportunities, Mobilizing Structures, and Cultural Framings*, edited by Doug McAdam, John D. McCarthy, and Mayer N. Zald. Cambridge, UK: Cambridge University Press.

McFarland, Daniel A. 2004. "Resistance as Social Drama: A Study of Change-Oriented Behaviors." *American Journal of Sociology* 109 (6):1249-1318.

McGrath, Cathleen, Jim Blythe, and David Krackhardt. 1997. "The Effect of Spatial Arrangements on Judgments and Errors in Interpreting Graphs." *Social Networks* 19:223-242.

McPherson, J. Miller, Lynn Smith-Lovin, and Matthew E. Brashears. 2006. "Social Isolation in America: Changes in Core Discussion Networks over Two Decades." *American Sociological Review* 71 (June):353-375.

Meyer, John W., John Boli, George M. Thomas, and Francisco O. Ramirez. 1997. "World Society and the Nation-State." *American Journal of Sociology* 103 (1):144-181.

Meyer, John W., and Brian Rowan. 1977. "Institutionalized Organizations: Formal Structure as Myth and Ceremony." *American Journal of Sociology* 83 (2):340-363.

Milgram, Stanley. 1967. "The Small-World Problem." *Psychology Today* 1 (May):61-67.

________. 1974. *Obedience to Authority*. Princeton, NJ: Princeton University Press.

Mitchell, J. Clyde (Ed.). 1969. *Social Networks in Urban Situations: Analyses of Personal Relationships in Central African Towns*. Manchester, England: Manchester University Press.

Moody, James. 2001. "Peer Influence Groups: Identifying Dense Clusters in Large Networks." *Social Networks* (23):261-283.

Moreno, Jacob L. 1953. *Who Shall Survive? Foundations of Sociometry, Group Psycholtherapy and Sociodrama*. Beacon, NY: Beacon House.

Mydans, Seth. 2008. "Nasir Abas, Terrorist Defector, Aids Indonesian Police." *New York Times*. Retrieved September 12, 2008, from the World Wide Web: http://www.iht.com/articles/2008/02/29/asia/profile.php

Nadel, Siegfried F. 1957. *The Theory of Social Structure*. London: Cohen and West.

Osa, Maryjane. 2003. "Networks of Opposition: Linking Organizations Through Activists in the Polish People's Republic." Pp. 77-104 in *Social Movements and Networks: Relational Approaches to Collective Action*, edited by Mario Diani and Doug McAdam. New York and Oxford: Oxford University Press.

Padgett, John F., and Christopher K. Ansell. 1993. "Robust Action and the Rise of the Medici, 1400-1434." *American Journal of Sociology* 98:1259-1319.

Pedahzur, Ami, and Arie Perliger. 2006. "The Changing Nature of Suicide Attacks: A Social Network Perspective." *Social Forces* 84 (4):1987-2008.

Pedersen, Hans, and Trond Gimmingsrud. 2008. "Strategies for Disrupting Noordin's Network." Monterey, California: Naval Postgraduate School.
Pescosolido, Bernice A., and Sharon Georgianna. 1989. "Durkheim, Suicide, and Religion: Toward a Network Theory of Suicide." *American Sociological Review* 54 (1):33-48.
Popielarz, Pamela A., and J. Miller McPherson. 1995. "On the Edge or in Between: Niche Position, Niche Overlap, and the Duration of Voluntary Association Memberships." *American Journal of Sociology* 101 (3):698-720.
Powell, Walter W., and Paul J. DiMaggio (Eds.). 1991. *The New Institutionalism in Organizational Analysis*. Chicago and London: University of Chicago Press.
Raab, Jorg, and H. Brinton Milward. 2003. "Dark Networks as Problems." *Journal of Public Administration Research and Theory* 13 (4):413-439.
Radcliffe-Brown, Alfred R. 1940. "On Social Structure." *Journal of the Rolyal Anthropological Society of Great Britain and Ireland* LXX:1-12.
Richardson, David C., and Jane S. Richardson. 1992. "The Kinemage: A Tool for Scientific Communication." *Protein Science* 1 (1):3-9.
Roberts, John M. Jr. 2000. "Correspondence Analysis of Two-mode Network Data." *Social Networks* 22:65-72.
Rodriguez, Jose' A. 2005. "The March 11th Terrorist Network: In Its Weakness Lies Its Strength." Presented at Sunbelt XXV: International Sunbelt Social Network Conference, February 16-21, Redondo Beach, CA
Roethlisberger, Fritz J., and William J. Dickson. 1939. *Management and the Worker*. Cambridge, MA: Harvard University Press.
Sageman, Marc. 2004a. "Statement to the National Commission on Terrorist Attacks Upon the United States." Retrieved, 2006, from the World Wide Web: http://www.9-11commission.gov/hearings/hearing3/witness_sageman.htm
________. 2004b. *Understanding Terror Networks*. Philadelphia, PA: University of Pennsylvania Press.
________. 2008. *Leaderless Jihad: Terror Networks in the Twenty-First Century*. Philadelphia: University of Pennsylvania Press.
Saxenian, AnnaLee. 1994. *Regional Advantage: Culture and Competition in Silicon Valley and Route 128*. Cambridge, MA: Harvard University Press.
________. 1996. "Inside-Out: Regional Networks and Industrial Adaptation in Silicon Valley and Route 128." *Cityscape: A Journal of Policy Development and Research* 2 (2):41-60.
Scott, John. 2000. *Social Network Analysis: A Handbook*. 2nd ed. Thousand Oaks, CA: Sage Publications.
Simmel, Georg. [1908, 1922] 1955. *Conflict & The Web of Group-Affiliations*. Translated by Kurt H. Wolff and Reinhard Bendix. New York: The Free Press.
Smith, Christian S. 1991. *The Emergence of Liberation Theology: Radical Religion and Social Movement Theory*. Chicago: University of Chicago Press.

________. 1996. *Resisting Reagan: The U.S. Central America Peace Movement*. Chicago: The University of Chicago Press.

Snow, David A., and Cynthia L. Phillips. 1980. "The Lofland-Stark Conversion Model: A Critical Assessment." *Social Problems* 27:430-447.

Snow, David A., Louis A. Zurcher, and Sheldon Ekland-Olson. 1980. "Social Networks and Social Movements: A Microstructural Approach to Differential Recruitment." *American Sociological Review* 45:787-801.

Sparrow, Malcom K. 1991. "The Application of Network Analysis to Criminal Intelligence: An Assessment of the Prospects." *Social Networks* 13:251-274.

Stark, Rodney. 1996. *The Rise of Christianity: A Sociologist Reconsiders History*. Princeton, NJ: Princeton University Press.

________. 2007. *Sociology*. 10th ed. Belmont, CA: Wadsworth Publishing Company.

Stark, Rodney, and William Sims Bainbridge. 1980. "Networks of Faith: Interpersonal Bonds and Recruitment to Cults and Sects." *American Journal of Sociology* 85 (6):1376-1395.

Stark, Rodney, edited by Reid L. Nielson. 2005. *The Rise of Mormonism*. New York: Columbia University Press.

Sunstein, Cass R. 2003. *Why Societies Need Dissent*. Cambridge, MA: Harvard University Press.

Travers, Jeffrey, and Stanley Milgram. 1969. "An Experimental Study of the Small World Problem." *Sociometry* 32 (4):425-443.

Tsvetovat, Maksim, and Kathleen M. Carley. 2005. "Structural Knowledge and Success of Anti-Terrorist Activity: The Downside of Structural Equivalence." Pp. No. 2 in *Journal of Social Structure*.

Turner, Jonathan H. 2006. *Sociology*. Upper Saddle River, NJ: Pearson Prentice Hall.

Uzzi, Brian. 1996. "The Sources and Consequences of Embeddedness for the Economic Performance of Organizations: The Network Effect." *American Sociological Review* 61 (4):674-698.

________. 2008. "A Social Network's Changing Statistical Properties and the Quality of Human Innovation." *Journal of Physics A: Mathematical and Theoretical* 41:1-12.

Uzzi, Brian, and Jarrett Spiro. 2005. "Collaboration and Creativity: The Small World Problem." *American Journal of Sociology* 111 (2):447-504.

Warner, W. Lloyd, and Paul S. Lunt. 1941. *The Social Life of a Modern Community*. New Haven, CT: Yale University Press.

Wasserman, Stanley, and Katherine Faust. 1994. *Social Network Analysis: Methods and Applications*. Cambridge, UK: Cambridge University Press.

Watts, Duncan J. 1999a. "Networks, Dynamics, and the Small-World Phenomenon." *American Journal of Sociology* 105 (2):493-527.

________. 1999b. *Small Worlds: The Dynamics of Networks Between Order and Randomness*. Princeton, NJ: Princeton University Press.

________. 2003. *Six Degrees: The Science of a Connected Age*. New York: W. W. Norton & Company.

Watts, Duncan J., Peter Sheridan Dodds, and M. E. Newman. 2003. "Identity and Search in Social Networks." *Science* 296:1302-1304.

White, Harrison C. 1992. *Identity and Control: A Structural Theory of Social Action*. Princeton, NJ: Princton University Press.

________. 2008. *Identity and Control: How Social Formations Emerge*. 2nd ed. Princeton, NJ: Princeton University Press.

White, Harrison C., Scott A. Boorman, and Ronald L. Breiger. 1976. "Social Structure from Multiple Networks I: Blockmodels of Roles and Positions." *American Journal of Sociology* 81:730-780.

Wikipedia. 2007. "Social Structure." *Wikipedia*. Retrieved July 18, 2007, from the World Wide Web: http://en.wikipedia.org/wiki/Social_structure

Wolfe, Alvin W. 1978. "The Rise of Network Thinking in Anthropology." *Social Networks* 1:53-64.

Zimbardo, Philip G., Christina Maslasch, and Craig Haney. 2000. "Reflections on the Stanford Prison Experiment: Genesis, Transformations, Consequences." Pp. 193-237 in *Obedience to Authority: Current Perspectives on the Milgram Paradigm*, edited by Thomas Blass. Mahwah, NJ: Lawrence Erlbaum Associates, Publishers.